DAG [Delivery and Assessment Guide]

Select A Round: 61 **Batch** CS/ACSL-M/61/01 Module --ALL-- Show

DAG

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
0			Introductory class on: • Concept of IT • Operating System (Windows 10)	06/06/2024	
0			Review Class	06/06/2024	
0			Introductory class on: • Concept of IT • Operating System (Windows 10)	08/06/2024	
0			Review Class	08/06/2024	

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
1	Module01	STC CS CIT 101	**Hardware Concepts (Reference courseware from this point: CONCEPTS OF IT, starting from page: 15); Computer performance; Memory & Storage; Input devices, Output devices; Software Concepts; Network Types; Data Transfer** 1. Understand the term hardware. 2. Understand personal computers. Distinguish between desktop, laptop (notebook), tablet PC, etc. in terms of typical use. 3. Identify common handheld portable digital devices like: personal digital assistant (PDA), mobile phone, smartphone, multimedia player and know their main features. 4. Know the main parts of a computer like: central processing unit (CPU), types of memory, hard disk, common input and output devices. 5. Identify common input/output ports like USB, serial, parallel, network port, FirmWire, etc. 6. Know some of the factors that impact computer's performance like: CPU speed, RAM size, graphics card processor and memory, the number of applications running. 7. Know that the speed (operating frequency) of the CPU is measured in megahertz (MHz) or gigahertz (GHz). 8. Understand what is computer memory, RAM, ROM, ROM-BIOS, Video Memory, storage capacity management, types of storage media 9. Know storage capacity measurements: bit, byte, KB, MB, GB, TB. 10. Know the main types of storage media like: internal hard disk, external hard disk, network drive, CD, DVD, USB flash drive, memory card, online file storage. 11. Identify some of the main input devices like: mouse, keyboard, trackball, scanner, touch pad, stylus, joystick, web camera (webcam), digital camera, microphone, etc. 12. Know some of the main output devices like: screens/monitors, printers, speakers, headphones. 13. Understand that some devices are both input and output devices like: touch screens. 14. Understand what software is. 15. Understand what an operating system is and name some common operating systems. 16. Identify and know the uses of some common software applications. 17. Distinguish between operating systems software and applications software. 18. Know some options available for enhancing accessibility. 19. Understand the terms local area network (LAN), wireless local area network (WLAN), wide area network (WAN). 20. Understand the term client/server. 21. Understand what the Internet is and know some of its main uses. 22. Understand what are intranet and extranet. 23. Understand the concepts of downloading from, and uploading to a network. 24. Understand what data transfer rate means. Understand how it is measured: bits per second (bps), kilobytes per second (kbps), megabits per second (mbps). 25. Know about different Internet connection services: dial-up, broadband. 26. Know about different options for connecting to the Internet. 27. Understand some of the characteristics of broadband: always-on, typically a flat fee, high speed, higher risk of intruder attack. **EVIDENCE GUIDANCE**	09/06/2024	09/06/2024

Sl	Module	Competency Unit	Elements		Schedule Date	Actual Date
			Critical aspects of evidence	**<u>Assessment must confirm the ability to</u> –** • **Understanding computer, devices, hardware, main parts and peripherals** • **Identify common input/output ports** • **Understand computer memory and stores** • **Know storage capacity measurements** • **Identify some of the main input and output devices** • **Understand software, OS, applications and there implications** • **Understand LAN, WLAN, WAN, Client/Server** • **Understand Internet and it's use, upload, download** • **Understand data transfer rate and it's measurement units** • **Understand different internet connection services, options and characteristics.**		
			Assessment of unit	This unit may be assessed by taking **<u>a written assessment</u>** and **<u>an evidence assessment</u>** after completion of the unit. The written assessment will contain **<u>20 MCQs</u>** and **<u>10 short descriptive questions</u>**. The evidence assessment will include **<u>3-4 practical questions</u>** incorporating the common tasks related to the competency unit. **<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant.**.		

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
2	Module01	STC CS CIT 102	**The Windows 10 Desktop; Manipulating Desktop Windows (Reference Courseware from this point: WINDOWS 10 Foundation, starting from page: 7); Knowing Windows 10 Desktop, Desktop Icons, Taskbar, Customize Desktop, Setting and modifying Desktop Background, Screen Saver, Date and Time, Other setting; Applications & the Desktop; Help in Windows 10; File Explorer; Manipulating folders; Manipulating files** 1. What is the Windows Desktop? 2. Knowing Desktop Icons 3. Know How to Move Desktop Icons 4. Knowing Windows Taskbar 5. Showing Hidden Icons 6. Setting Taskbar Clock 7. Modifying the Computer Date and Time 8. Setting the Keyboard Language and Other Regional Setting 9. Know how to Drag and Drop Desktop icons, files and folder 10. Know how to select and manipulate 11. Single Clicking vs. Double Clicking 12. Moving a Window on the Desktop 13. Resizing a Window, Narrower or Wider 14. Resizing a Window, Taller or Shorter 15. Resizing a Window in Two Directions at Once 16. Maximizing and Minimizing Window 17. Closing a Window 18. The Start Button and Start Menu 19. Multitasking 20. How to start an application 21. Finding Applications such as WordPad 22. Using Print Screen Within WordPad 23. Identifying the Parts of an Application Window 24. Title Bar, The Ribbon, Ribbon Tabs 25. Status Bar 26. Installing Programs 27. Removing Program 28. Shutting Down the Computer 29. Restarting the Computer 30. Using the search for Help Box 31. Displaying Basic Information About Your PC 32. Knowing about Files and Folders 33. Type of Drive and Drive Letters 34. Opening the Windows File Explorer Program 35. Displaying or Hiding the Ribbon 36. The File Explorer Windows 37. File Explorer - Folders 38. File Explorer - Devices 39. File Explorer - Network Locations 40. File Explorer - Downloads 41. File Explorer - Desktop 42. File Explorer - OneDrive 43. File Explorer - Homegroup 44. File Explorer - This PC 45. Displaying the Contents of a Particular Folder 46. Changing the View Within the File Explorer 47. Changing the Sort Order in Details View 48. Drives, Folders and Files 49. Data Storage Device 50. File sizes and Disk Storage Capacities 51. Creating Folder 52. Creating a Sub-Folder 53. File Naming Issues 54. Renaming a Folder 55. Deleting a Folder 56. Displaying Folder Details 57. File Types 58. Selecting Multiple files 59. Counting files in a Folder 60. File Attributes 61. Renaming Files 62. Deleting Files 63. Restoring Items From the Recycle Bin 64. Emptying the Recycle Bin 65. Permanently Deleting a File (Bypassing the Recycle Bin) 66. Recycled large files 67. The Windows Clipboard	10/06/2024	10/06/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>
68. Moving Files Between Folders
69. Copying Files Between Folders
70. Backup
71. Removable Memory Sticks – Auto play
72. Backing up data to a removable Drive
73. Safely Ejecting a Memory Stick
74. What is File Compression?
75. Compressing Files
76. Extracting Compressed Files
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Understanding Windows Desktop
Handling Desktop Icons
Handling Windows Taskbar and Icons
Modifying the Computer Date and Time
Setting the Keyboard Language and Other Regional Setting
Know how to drag and drop Desktop icons, files and folder
Understand the uses of single click and double click
Handle Window
Using start button and start menu
Multitasking
Finding and using application. Identifying the Pprts of an application window (such as - Title Bar, The Ribbon, Ribbon Tabs, Status Bar)
Installing and Removing Programs
Shutting Down and restarting the Computer
Searching for help. Displaying basic information about Your PC
Using files, folders and drivers
Using file explorer windows
Managing Recycle Bin. Bypassing the Recycle Bin
Using windows clipboard
Backup, Backing up data to a removable Drive, File Compression, Extracting Compressed Files</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>a written assessment</u> and <u>an evidence assessment</u> after completion of the unit.
The written assessment will contain <u>20 MCQs</u> and <u>10 short descriptive questions</u>.
The evidence assessment will include <u>3-4 practical questions</u> incorporating the common tasks related to the competency unit.

<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant..</td></tr>
</table>
</td><td></td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
3	Module01	STC CS CIT 103	**The Internet (Reference courseware from this point: WEB BROWSING & COMMUNICATION, starting from page: 697); Basic Browsing; Settings; Navigation; Bookmarks; E-mail Theory: Gmail** 1. Internet 2. World Wide Web (WWW) 3. Web sites and URLs 4. URL (Uniform Resource Locator) 5. Hyperlinks 6. ISP (Internet Service Provider) 7. Structure of a Web Address 8. Web Browser 9. Search Engines 10. Really Simple Syndication (RSS) feeds 11. Podcasts 12. Opening the Browser 13. Minimizing, Maximizing, Restoring and Closing 14. Entering a URL into the Address Bar 15. Creating a new tab 16. Stop button 17. Forcing a Web page to display within a new window 18. Forcing a Web page to display within a new tab 19. Reloading/Refreshing Web 20. Switching between tabs 21. Closing a tab 22. Other Firefox icons 23. Firefox menu bar 24. Customize Firefox controls, buttons and toolbars 25. Setting Home Page (Start Page) 26. Setting multiple Home Pages 27. Browsing history in Firefox - View the websites you have visited 28. Deleting a history item 29. What things are included in history 30. How to clear history 31. Clear my history automatically 32. Remove a single website from your history 33. What are Pop-ups 34. Pop-up blocker settings 35. Pop-ups not being blocked 36. Hyperlinks 37. Back and Forward 38. Navigating through Web sites 39. Revisiting Web pages via the address bar 40. How to bookmark a page? 41. How to change the name or location of a bookmark? 42. How to find bookmarks 43. How to add the Bookmarks Menu button to the toolbar 44. How to organize bookmarks 45. How to turn on the Bookmarks Toolbar 46. Deleting a single bookmark 47. Deleting more than one bookmark or folder 48. Opening gmail interface 49. Creating a gmail account 50. How to login and logout within gmail 51. Introducing gmail like Inbox, Draft, Sent and spam etc. 52. Use of reply, reply to all, forward 53. How to create (compose) and send an email 54. Clear understanding of “To, CC, BCC” of sending an email 55. How to attach file from gmail 56. How to remove attached file with gmail 57. Marking message as “read” and “unread” 58. How to save attachment of incoming message 59. How to trash a message 60. How to delete permanently form trash/bin **EVIDENCE GUIDANCE**	11/06/2024	11/06/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Understanding and using Internet, World Wide Web (WWW), Web sites and URLs, Hyperlinks, Structure of a Web Address.
Understanding and using Web Browser, Search Engines, RSS Feed, Podcasts.
Using and Managing web browsers, browsing histories, bookmark, hadnling Pop-ups.
Opening, creating, managing and using gmail account.</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>a written assessment</u> and <u>an evidence assessment</u> after completion of the unit.
The written assessment will contain <u>20 MCQs</u> and <u>10 short descriptive questions</u>.
The evidence assessment will include <u>3-4 practical questions</u> incorporating the common tasks related to the competency unit.

<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant..</td></tr>
</table>
</td><td></td><td></td></tr>
<tr><td>4</td><td>Module01</td><td>STC CS CIT 104</td><td>
The Microsoft Word 2016 Screen; Starting to use Microsoft Word 2016; Using Templates to Create Documents (Reference courseware from this point: MICROSOFT WORD 2016 FOUNDATION starting from page: 06); Using the book “Microsoft Word 2016 Foundation”

Starting Word 2016
Opening a blank document
Introducing with File tab, Ribbon tabs and groups, Dialog Box Launcher
Using the default Microsoft Word document
Saving Microsoft Word documents
Opening and Closing documents
Save As
Creating a new blank document using a Keyboard shortcut
Using help within Microsoft Word
Using 'TELL ME' within Microsoft Word
ALT key help
Closing Microsoft Word
Creating new documents using Different Templates
Using online templates

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
After completing this competency unit trainees will be able to create reach Word document.</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>a written assessment</u> and <u>an evidence assessment</u> after completion of the unit.
The written assessment will contain <u>20 MCQs</u> and <u>10 short descriptive questions</u>.
The evidence assessment will include <u>3-4 practical questions</u> incorporating the common tasks related to the competency unit.

<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant..</td></tr>
</table>
</td><td>12/06/2024</td><td>12/06/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>5</td><td>Module01</td><td>STC CS CIT 104</td><td>Manipulating text; Microsoft Word 2016 text formatting; Microsoft Word 2016 paragraph formatting; Borders and Shading within Microsoft Word 2016
1. Select, then format
2. Selecting text
3. Inserting, deleting, undo and redo
4. Copying text within a document
5. Moving (cutting) text within a document
6. What is text formatting?
7. Font: type, size, increase, decrease, bold, italic, underline, subscript, superscript
8. Case changing
9. Highlighting
10. Font color
11. Copying text formatting, removing formatting
12. Using zoom
13. Inserting special characters and symbols
14. What is paragraph formatting
15. Paragraph marks, line break
16. Techniques for aligning and indenting text
17. Appling paragraph spacing (single, double, above, below)
18. Using paragraph spacing rather than using the return key
19. Applying bullets to a list, number to alist
20. Modifying and removing bullet and numbering formatting
21. Adding and modifying borders
22. Adding and modifying shading
23. Applying borders to selected text
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• After completing this competency unit trainees will be able to create reach Word document.</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>a written assessment</u> and <u>an evidence assessment</u> after completion of the unit.
The written assessment will contain <u>20 MCQs</u> and <u>10 short descriptive questions</u>.
The evidence assessment will include <u>3-4 practical questions</u> incorporating the common tasks related to the competency unit.

<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant..</td></tr></table></td><td>13/06/2024</td><td>13/06/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>6</td><td>Module01</td><td>STC CS CIT 104</td><td>Microsoft Word 2016 styles; Using Word 2016 design themes; Microsoft Word 2016 page formatting; Word 2016 views and document navigation
1. Know about style and apply it
2. Applying design them and it color
3. Apply a customized font
4. What is page formatting?
5. Page orientation, page size and page margin
6. Insert and delete page breaks
7. Creating, editing header footer
8. Inserting page number
9. Cover pages
10. Applying automatic hyphenation
11. Switching between word views
12. Using the zoom tool
13. Navigating through documents

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• After completing this competency unit trainees will be able to create reach Word document.

Assessment of unit: This unit may be assessed by taking <u>a written assessment</u> and <u>an evidence assessment</u> after completion of the unit.
The written assessment will contain <u>20 MCQs</u> and <u>10 short descriptive questions</u>.
The evidence assessment will include <u>3-4 practical questions</u> incorporating the common tasks related to the competency unit.

<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant..</td><td>22/06/2024</td><td>22/06/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>7</td><td>Module01</td><td>STC CS CIT 104</td><td>Tables; Using graphics within Microsoft Word; Finding and Replacing text within a Microsoft 2016 document; Using multiple open documents
1. Using tables
2. Inserting a table
3. Navigating within a table
4. Selecting and editing text within a table
5. Selecting cells, rows, columns or the entire table
6. Inserting and deleting rows and columns
7. Modifying column width or row height
8. Modifying the table width
9. Modifying table styles
10. Inserting pictures, Online pictures, Shapes, SmartArt, Chart
11. Modifying the Chart type, Chart style, Chart data
12. Inserting a screenshot
13. Selecting, resizing and deleting graphics
14. Copying or moving graphics
15. Using find and replace find the text and replace the text
16. Multitasking with Word 2016
17. Switching between open documents
18. Tiling or cascading documents on your screen
19. Comparing documents side by side
20. Copying or moving selected items between documents
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• After completing this competency unit trainees will be able to create reach Word document.</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>a written assessment</u> and <u>an evidence assessment</u> after completion of the unit.
The written assessment will contain <u>20 MCQs</u> and <u>10 short descriptive questions</u>.
The evidence assessment will include <u>3-4 practical questions</u> incorporating the common tasks related to the competency unit.

<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant..</td></tr></table></td><td>23/06/2024</td><td>23/06/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>8</td><td>Module01</td><td>STC CS CIT 105</td><td>Getting started with Excel 2016; Excel 2016 selection techniques, Manipulating rows and columns within excel 2016; Manipulating cells and cell content within excel 2016; Excel 2016 worksheets; Font formatting within Excel 2016 (Reference courseware from this point: MICROSOFT Excel 2016 FOUNDATION starting from page: 08); Using the book “Microsoft Excel 2016 Foundation”
1. Selecting the blank worksheet template
2. Entering text, number and date
3. Alignment and editing of text, number and date
4. Creating, saving, opening and closing excel 2016
5. Saving a workbook using a different file type and different name
6. Different type of selection techniques
7. Inserting and deleting rows and columns into a worksheet
8. Modifying column widths and row heights
9. Copying, moving a cell or range contents within a workbook
10. Editing and deleting cell content
11. Indo and redo
12. Copying and moving data between worksheets (within the same workbook)
13. Moving data worksheets (between different workbooks)
14. Copying data between worksheets (in different workbooks)
15. Autofill
16. Sorting a cell range
17. Switching between worksheets
18. Recommended techniques with naming worksheets
19. Inserting and renaming a new worksheet
20. Copying, moving a worksheet within a workbook and between workbooks
21. Different types of font formatting
22. Working with different type of
23. Text wrapping format painter

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• After completing this competency unit trainees will be able to do enterprise level and critical spread sheet works.

Assessment of unit: This unit may be assessed by taking <u>a written assessment</u> and <u>an evidence assessment</u> after completion of the unit.
The written assessment will contain <u>20 MCQs</u> and <u>10 short descriptive questions</u>.
The evidence assessment will include <u>3-4 practical questions</u> incorporating the common tasks related to the competency unit.

<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant..</td><td>24/06/2024</td><td>24/06/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>9</td><td>Module01</td><td>STC CS CIT 105</td><td>Alignment formatting within Excel 2016; Number formatting within Excel 2016; Freezing row and column titles within Excel 2016
1. Working with different type of alignment
2. Centering a title over a cell range
3. Cell orientation
4. Text wrapping
5. Use of format painter
6. Number formatting: decimal point display, comma, currency
7. Date Styles
8. Percentage
9. Freezing row and column titles
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• After completing this competency unit trainees will be able to do enterprise level and critical spread sheet works.</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>a written assessment</u> and <u>an evidence assessment</u> after completion of the unit.
The written assessment will contain <u>20 MCQs</u> and <u>10 short descriptive questions</u>.
The evidence assessment will include <u>3-4 practical questions</u> incorporating the common tasks related to the competency unit.

<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant..</td></tr></table></td><td>25/06/2024</td><td>25/06/2024</td></tr>
<tr><td>10</td><td>Module01</td><td>STC CS CIT 105</td><td>Excel 2016 formulas; Excel 2016 functions
1. Crating and copying formula
2. Using operator in formula
3. Relative and absolute cell referencing within formulas
4. What is function
5. Working with different types of functions
6. WHAT ARE 'IF FUNCTIONS'?
7. Using the IF FUNCTION
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• After completing this competency unit trainees will be able to do enterprise level and critical spread sheet works.</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>a written assessment</u> and <u>an evidence assessment</u> after completion of the unit.
The written assessment will contain <u>20 MCQs</u> and <u>10 short descriptive questions</u>.
The evidence assessment will include <u>3-4 practical questions</u> incorporating the common tasks related to the competency unit.

<u>n.b.</u>- Do not forget to email the question scripts (along with answer-keys marked in bold for MCQs) to the concerned consultant..</td></tr></table></td><td>26/06/2024</td><td>26/06/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>11</td><td>Module02</td><td>STC CS SQL-201</td><td>Set up computer for SQl Server 2016 and an introduction to relational databases and SQL
1. Three editions of SQL Server 2016
2. The tool for working with all editions of SQL Server
3. How to install SQL Server 2016
4. How to install SQL Server Management Studio
5. How to install the files for this book
6. How to create the databases for this book
7. How to restore the databases for this book
8. How to install Visual Studio
9. An introduction to client/server systems
a. The hardware components of a client/server system
b. The software components of a client/server system
c. Other client/server system architectures
10. An introduction to the relational database model
a. How a database table is organized
b. How the tables in a relational database are related
c. How the columns in a table are defined
d. How relational databases compare to other data models
11. An introduction to SQL and SQL-based systems
a. A brief history of SQL
b. A comparison of Oracle, DB2, MySQL, and SQL Server
12. The Transact-SQL statements
a. An introduction to the SQL statements
b. Typical statements for working with database objects
c. How to query a single table
d. How to join data from two or more tables
e. How to add, update, and delete data in a table
f. SQL coding guidelines
13. How to work with other database objects
a. How to work with views
b. How to work with stored procedures, triggers, and user-defined functions
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Install SQL Server 2016, SQL Server Management
• Use the tool for working with all editions of SQL Server
• Understand the client/server
• Understand the relational database model
• Writing Transact-SQL statements
• Working with views, stored procedures, triggers, and user-defined functions</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>27/06/2024</td><td>27/06/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Review</td><td>29/06/2024</td><td></td></tr>
<tr><td>0</td><td></td><td></td><td>Review</td><td>30/06/2024</td><td></td></tr>
<tr><td>0</td><td></td><td></td><td>Review</td><td>01/07/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>12</td><td>Module02</td><td>STC CS SQL-202</td><td>
How to use the Management Studio
1. An introduction to SQL Server 2016
a. A summary of the SQL Server 2016 tools
b. How to start and stop the database engine
c. How to enable remote connections
2. An introduction to the Management Studio
a. How to connect to a database server
b. How to navigate through the database objects
3. How to manage the database files
a. How to attach a database
b. How to detach a database
c. How to back up a database
d. How to restore a database
e. How to set the compatibility level for a database
4. How to view and modify the database
a. How to view the database diagrams
b. How to view the column definitions of a table
c. How to modify the column definitions
d. How to view the data of a table
e. How to modify the data of a table
5. How to work with queries
a. How to enter and execute a query
b. How to handle syntax errors
c. How to open and save queries
d. An introduction to the Query Designer
6. How to view the documentation for SQL Server
a. How to display the MSDN website
b. How to look up information on the MSDN website
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Use SQL Server Management Studio and it's all features.</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>02/07/2024</td><td>02/07/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>13</td><td>Module02</td><td>STC CS SQL-203</td><td>
How to retrieve data from a single table

An introduction to the SELECT statement
<ol type="a">
The basic syntax of the SELECT statement
SELECT statement examples

How to code the SELECT clause
<ol type="a">
How to code column specifications
How to name the columns in a result set
How to code string expressions
How to code arithmetic expressions
How to use functions
How to use the DISTINCT keyword to eliminate duplicate rows
How to use the TOP clause to return a subset of selected rows

How to code the WHERE clause
<ol type="a">
How to use comparison operators
How to use the AND, OR and NOT logical operators
How to use the IN operator
How to use the BETWEEN operator
How to use the LIKE operator
How to use the IS NULL clause

How to code the ORDER BY clause
<ol type="a">
How to sort a result set by a column name
How to sort a result set by an alias, an expression or a column number
How to retrieve a range of selected rows

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –

Query a single table by using SELECT, DISTINCT, TOP clause
Apply condition by using WHERE clause and AND, OR, NOT, IN, BETWEEN, LIKE operators
Use IS NULL clause
Sort result set by using ORDER BY clause.
</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>03/07/2024</td><td>03/07/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>14</td><td>Module02</td><td>STC CS SQL-204</td><td>How to retrieve data from two or more tables
1. How to work with inner joins
a. How to code an inner join
b. When and how to use correlation names
c. How to work with tables from different databases
d. How to use compound join conditions
e. How to use a self-join
f. Inner joins that join more than two tables
g. How to use the implicit inner join syntax
2. How to work with outer joins
a. How to code an outer join
b. Outer join examples
c. Outer joins that join more than two tables
3. Other skills for working with joins
a. How to combine inner and outer joins
b. How to use cross joins
4. How to work with unions
a. The syntax of a union
b. Unions that combine data from different tables
c. Unions that combine data from the same table
d. How to use the EXCEPT and INTERSECT operators
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Work with inner join, outer join, full join, cross join and union</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>04/07/2024</td><td>04/07/2024</td></tr>
<tr><td>15</td><td>Module02</td><td>STC CS SQL-205</td><td>How to code summary queries
1. How to work with aggregate functions
a. How to code aggregate functions
b. Queries that use aggregate functions
2. How to group and summarize data
a. How to code the GROUP BY and HAVING clauses
b. Queries that use the GROUP BY and HAVING clauses
c. How the HAVING clause compares to the WHERE clause
d. How to code complex search conditions
3. How to summarize data using SQL Server extensions
a. How to use the ROLLUP operator
b. How to use the CUBE operator
c. How to use the GROUPING SETS operator
d. How to use the OVER clause
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Work with aggregate functions
• Group and summarize data using GROUP BY and HAVING clauses
• Summarize data using ROLLUP, CUBE, GROUPING SETS operators and OVER clause</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>06/07/2024</td><td>06/07/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>16</td><td>Module02</td><td>STC CS SQL-206</td><td>How to code subqueries
1. An introduction to subqueries
a. How to use subqueries
b. How subqueries compare to joins
2. How to code subqueries in search conditions
a. How to use subqueries with the IN operator
b. How to compare the result of a subquery with an expression
c. How to use the ALL keyword
d. How to use the ANY and SOME keywords
e. How to code correlated subqueries
f. How to use the EXISTS operator
3. Other ways to use subqueries
a. How to code subqueries in the FROM clause
b. How to code subqueries in the SELECT clause
4. Guidelines for working with complex queries
a. A complex query that uses subqueries
b. A procedure for building complex queries
5. How to work with common table expressions
a. How to code a CTE
b. How to code a recursive CTE

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Code subqueries and correlated subqueries
• Code subqueries in the FORM and SELECT clauses
• Use subqueries with IN, ALL, ANY, SOME, EXISTS operators
• Work with CTE
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>07/07/2024</td><td>07/07/2024</td></tr>
<tr><td>17</td><td>Module02</td><td>STC CS SQL-207</td><td>How to insert, update, and delete data
1. How to create test tables
a. How to use the SELECT INTO statement
b. How to use a copy of the database
2. How to insert new rows
a. How to insert a single row
b. How to insert multiple rows
c. How to insert default values and null values
d. How to insert rows selected from another table
3. How to modify existing rows
a. How to perform a basic update operation
b. How to use subqueries in an update operation
c. How to use joins in an update operation
4. How to delete existing rows
a. How to perform a basic delete operation
b. How to use subqueries and joins in a delete operation
5. How to merge rows
a. How to perform a basic merge
b. How to code more complex merge operations

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Perform complex nd compound insert, update, delete and merge operations
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>08/07/2024</td><td>08/07/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>18</td><td>Module02</td><td>STC CS SQL-208</td><td>How to work with data types
1. A review of the SQL data types
a. Data type overview
b. The numeric data types
c. The string data types
d. The date/time data types
e. The large value data types
2. How to convert data
a. How data conversion works
b. How to convert data using the CAST function
c. How to convert data using the CONVERT function
d. How to use the TRY CONVERT function
e. How to use other data conversion functions
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand and use SQL data types
• Convert data from one type to another type</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>09/07/2024</td><td>10/07/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Orientation</td><td>09/07/2024</td><td></td></tr>
<tr><td>19</td><td>Module02</td><td>STC CS SQL-209</td><td>How to work with functions
1. How to work with string data
a. A summary of the string functions
b. How to solve common problems that occur with string data
2. How to work with numeric data
a. A summary of the numeric functions
b. How to solve common problems that occur with numeric data
3. How to work with date/time data
a. A summary of the date/time functions
b. How to parse dates and times
c. How to perform operations on dates and times
d. How to perform a date search
e. How to perform a time search
4. Other functions you should know about
a. How to use the CASE function
b. How to use the IIF and CHOOSE functions
c. How to use the COALESCE and ISNULL functions
d. How to use the GROUPING function
e. How to use the ranking functions
f. How to use the analytic functions
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Work with string, numeric and date/time data types
• Use string, numeric, date/time, ranking, analytic functions and other functions like - CASE, IIF, CHOOSE, COALESCE, ISNULL, GROUPING etc.</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>10/07/2024</td><td>11/07/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>20</td><td>Module02</td><td>STC CS SQL-209</td><td>How to work with functions
1. How to work with string data
a. A summary of the string functions
b. How to solve common problems that occur with string data
2. How to work with numeric data
a. A summary of the numeric functions
b. How to solve common problems that occur with numeric data
3. How to work with date/time data
a. A summary of the date/time functions
b. How to parse dates and times
c. How to perform operations on dates and times
d. How to perform a date search
e. How to perform a time search
4. Other functions you should know about
a. How to use the CASE function
b. How to use the IIF and CHOOSE functions
c. How to use the COALESCE and ISNULL functions
d. How to use the GROUPING function
e. How to use the ranking functions
f. How to use the analytic functions
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Work with string, numeric and date/time data types
• Use string, numeric, date/time, ranking, analytic functions and other functions like - CASE, IIF, CHOOSE, COALESCE, ISNULL, GROUPING etc.</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>11/07/2024</td><td>13/07/2024</td></tr>
<tr><td>21</td><td>Module02</td><td>STC CS SQL-210</td><td>How to design a database
1. How to design a data structure
a. The basic steps for designing a data structure
b. How to identify the data elements
c. How to subdivide the data elements
d. How to identify the tables and assign columns
e. How to identify the primary and foreign keys
f. How to enforce the relationships between tables
g. How normalization works
h. How to identify the columns to be indexed
2. How to normalize a data structure
a. The seven normal forms
b. How to apply the first normal form
c. How to apply the second normal form
d. How to apply the third normal form
e. When and how to denormalize a data structure
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand data structure
• Design data structure
• Normalize data structure
• Denormalize data structure</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>13/07/2024</td><td>16/07/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>22</td><td>Module02</td><td>STC CS SQL-210</td><td>How to design a database
1. How to design a data structure
a. The basic steps for designing a data structure
b. How to identify the data elements
c. How to subdivide the data elements
d. How to identify the tables and assign columns
e. How to identify the primary and foreign keys
f. How to enforce the relationships between tables
g. How normalization works
h. How to identify the columns to be indexed
2. How to normalize a data structure
a. The seven normal forms
b. How to apply the first normal form
c. How to apply the second normal form
d. How to apply the third normal form
e. When and how to denormalize a data structure
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand data structure
• Design data structure
• Normalize data structure
• Denormalize data structure</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>14/07/2024</td><td>10/08/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Preparatory leave Exam#01</td><td>14/07/2024</td><td></td></tr>
<tr><td>23</td><td>Module02</td><td>STC CS SQL-210</td><td>How to design a database
1. How to design a data structure
a. The basic steps for designing a data structure
b. How to identify the data elements
c. How to subdivide the data elements
d. How to identify the tables and assign columns
e. How to identify the primary and foreign keys
f. How to enforce the relationships between tables
g. How normalization works
h. How to identify the columns to be indexed
2. How to normalize a data structure
a. The seven normal forms
b. How to apply the first normal form
c. How to apply the second normal form
d. How to apply the third normal form
e. When and how to denormalize a data structure
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand data structure
• Design data structure
• Normalize data structure
• Denormalize data structure</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>15/07/2024</td><td>11/08/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>MonthlyEXAM</td><td>15/07/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>24</td><td>Module02</td><td>STC CS SQL-210</td><td>
How to design a database

How to design a data structure
<ol type="a">
The basic steps for designing a data structure
How to identify the data elements
How to subdivide the data elements
How to identify the tables and assign columns
How to identify the primary and foreign keys
How to enforce the relationships between tables
How normalization works
How to identify the columns to be indexed

How to normalize a data structure
<ol type="a">
The seven normal forms
How to apply the first normal form
How to apply the second normal form
How to apply the third normal form
When and how to denormalize a data structure

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Understand data structure
Design data structure
Normalize data structure
Denormalize data structure</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>16/07/2024</td><td>12/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>25</td>
<td>Module02</td>
<td>STC CS SQL-211</td>
<td>
How to create a database and its tables with SQL Statements

1. An introduction to DDL
a. The SQL statements for data definition
b. Rules for coding object names
2. How to create databases, tables, and indexes
a. How to create a database
b. How to create a table
c. How to create an index
d. How to use snippets to create database objects
3. How to use constraints
a. An introduction to constraints
b. How to use check constraints
c. How to use foreign key constraints
4. How to change databases and tables
a. How to delete an index, table, or database
b. How to alter a table
5. How to work with sequences
a. How to create a sequence
b. How to use a sequence
c. How to delete a sequence
d. How to alter a sequence
6. The script used to create the AP database
a. How the script works
b. How the DDL statements work

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Create, use, alter and delete database and objects, like- table, index, constraints, sequence
• Understand how script and DDL statements work</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>18/07/2024</td>
<td>13/08/2024</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>26</td><td>Module02</td><td>STC CS SQL-211</td><td>How to create a database and its tables with SQL Statements
1. An introduction to DDL
a. The SQL statements for data definition
b. Rules for coding object names
2. How to create databases, tables, and indexes
a. How to create a database
b. How to create a table
c. How to create an index
d. How to use snippets to create database objects
3. How to use constraints
a. An introduction to constraints
b. How to use check constraints
c. How to use foreign key constraints
4. How to change databases and tables
a. How to delete an index, table, or database
b. How to alter a table
5. How to work with sequences
a. How to create a sequence
b. How to use a sequence
c. How to delete a sequence
d. How to alter a sequence
6. The script used to create the AP database
a. How the script works
b. How the DDL statements work
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Create, use, alter and delete database and objects, like- table, index, constraints, sequence
• Understand how script and DDL statements work
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>10/08/2024</td><td>14/08/2024</td></tr>
<tr><td>27</td><td>Module02</td><td>STC CS SQL-212</td><td>How to create a database and its tables with the Management Studio
1. How to work with a database
a. How to create a database
b. How to delete a database
2. How to work with tables
a. How to create, modify, or delete a table
b. How to work with foreign key relationships
c. How to work with indexes and keys
d. How to work with check constraints
e. How to examine table dependencies
3. How to generate scripts
a. How to generate scripts for databases and tables
b. How to generate a change script when you modify a table
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Create, use, alter and delete database and objects, like- table, index, constraints, sequence using Management Studio
• Generate and change script
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>11/08/2024</td><td>15/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>28</td><td>Module02</td><td>STC CS SQL-213</td><td>
How to work with views

An introduction to views
<ol type="a">
How views work
Benefits of using views
How to create and manage views
<ol type="a">
How to create a view
Examples that create views
How to create an updatable view
How to delete or modify a view
How to use views
<ol type="a">
How to update rows through a view
How to insert rows through a view
How to delete rows through a view
How to use the catalog views
How to use the View Designer
<ol type="a">
How to create or modify a view
How to delete a view

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Understand how view works and denefits of using view
Create, use, alter and delete view
Create, modify and delete view using the View Designer</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>12/08/2024</td><td>17/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>29</td><td>Module02</td><td>STC CS SQL-214</td><td>
How to code scripts
1. An introduction to scripts
a. How to work with scripts
b. The Transact-SQL statements for script processing
2. How to work with variables and temporary tables
a. How to work with scalar variables
b. How to work with table variables
c. How to work with temporary tables
d. A comparison of the five types of Transact-SQL table
3. How to control the execution of a script
a. How to perform conditional processing
b. How to test for the existence of a database object
c. How to perform repetitive processing
d. How to use a cursor
e. How to handle errors
f. How to use surround-with snippets
4. Advanced scripting techniques
a. How to use the system functions
b. How to change the session settings
c. How to use dynamic SQL
d. A script that summarizes the structure of a database
e. How to use the SQLCMD utility
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand how to work with script and how to process Transact-SQL statements for script
• How to work with scalar variables, table variables and temporary tables
• How to control the execution of a script
• Using system functions, dynamic SQL, SQLCMD utility
• Changing session settings
• summarizing structure of a database using script</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>13/08/2024</td><td>18/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>30</td><td>Module02</td><td>STC CS SQL-215</td><td>
How to code stored procedures, functions, and triggers
1. Procedural programming options in Transact-SQL
a. Scripts
b. Stored procedures, user-defined functions, and triggers
2. How to code stored procedures
a. An introduction to stored procedures
b. How to create a stored procedure
c. How to declare and work with parameters
d. How to call procedures with parameters
e. How to work with return values
f. How to validate data and raise errors
g. A stored procedure that manages insert operations
h. How to pass a table as a parameter
i. How to delete or change a stored procedure
j. How to work with system stored procedures
3. How to code user-defined functions
a. An introduction to user-defined functions
b. How to create a scalar-valued function
c. How to create a simple table-valued function
d. How to create a multi-statement table-valued function
e. How to delete or change a function
4. How to code triggers
a. How to create a trigger
b. How to use AFTER triggers
c. How to use INSTEAD OF triggers
d. How to use triggers to enforce data consistency
e. How to use triggers to work with DDL statements
f. How to delete or change a trigger
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• creating, updating and using different types of stored procedures, user-defined functions and triggers</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>14/08/2024</td><td>19/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>31</td><td>Module02</td><td>STC CS SQL-215</td><td>
How to code stored procedures, functions, and triggers
1. Procedural programming options in Transact-SQL
a. Scripts
b. Stored procedures, user-defined functions, and triggers
2. How to code stored procedures
a. An introduction to stored procedures
b. How to create a stored procedure
c. How to declare and work with parameters
d. How to call procedures with parameters
e. How to work with return values
f. How to validate data and raise errors
g. A stored procedure that manages insert operations
h. How to pass a table as a parameter
i. How to delete or change a stored procedure
j. How to work with system stored procedures
3. How to code user-defined functions
a. An introduction to user-defined functions
b. How to create a scalar-valued function
c. How to create a simple table-valued function
d. How to create a multi-statement table-valued function
e. How to delete or change a function
4. How to code triggers
a. How to create a trigger
b. How to use AFTER triggers
c. How to use INSTEAD OF triggers
d. How to use triggers to enforce data consistency
e. How to use triggers to work with DDL statements
f. How to delete or change a trigger
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• creating, updating and using different types of stored procedures, user-defined functions and triggers</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>15/08/2024</td><td>20/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>32</td><td>Module02</td><td>STC CS SQL-216</td><td>How to manage transactions and locking
1. How to work with transactions
a. How transactions maintain data integrity
b. SQL statements for handling transactions
c. How to work with nested transactions
d. How to work with save points
2. An introduction to concurrency and locking
a. How concurrency and locking are related
b. The four concurrency problems that locks can prevent
c. How to set the transaction isolation level
3. How SQL Server manages locking
a. Lockable resources and lock escalation
b. Lock modes and lock promotion
c. Lock mode compatibility
4. How to prevent deadlocks
a. Two transactions that deadlock
b. Coding techniques that prevent deadlocks
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understand Transaction, Lock, Lockable resources, Lock escalation and Deadlocks
• Work with nested transactions and save points
• Prevent deadlocks
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>17/08/2024</td><td>21/08/2024</td></tr>
<tr><td>33</td><td>Module02</td><td>STC CS SQL-216</td><td>How to manage transactions and locking
1. How to work with transactions
a. How transactions maintain data integrity
b. SQL statements for handling transactions
c. How to work with nested transactions
d. How to work with save points
2. An introduction to concurrency and locking
a. How concurrency and locking are related
b. The four concurrency problems that locks can prevent
c. How to set the transaction isolation level
3. How SQL Server manages locking
a. Lockable resources and lock escalation
b. Lock modes and lock promotion
c. Lock mode compatibility
4. How to prevent deadlocks
a. Two transactions that deadlock
b. Coding techniques that prevent deadlocks
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understand Transaction, Lock, Lockable resources, Lock escalation and Deadlocks
• Work with nested transactions and save points
• Prevent deadlocks
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>18/08/2024</td><td>22/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>34</td><td>Module02</td><td>STC CS SQL-217</td><td>
How to manage database security

How to work with SQL Server login IDs
<ol type="a">
An introduction to SQL Server security
How to change the authentication mode
How to create login IDs
How to change login IDs or passwords
How to work with database users
How to work with schemas

How to work with permissions
<ol type="a">
How to grant or revoke object permissions
The SQL Server object permissions
How to grant or revoke schema permissions
How to grant or revoke database permissions
How to grant or revoke server permissions

How to work with roles
<ol type="a">
How to work with the fixed server roles
How to work with user-defined server roles
How to display information about server roles and role members
How to work with the fixed database roles
How to work with user-defined database roles
How to display information about database roles and role members
How to deny permissions granted by role membership
How to work with application roles

How to manage security using the Management Studio
<ol type="a">
How to work with login IDs
How to work with the server roles for a login ID
How to assign database access and roles by login ID
How to assign user permissions to database objects
How to work with database permissions

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Work with SQL Server security, login IDs, passwords, users, schemas, permissions, roles and memberships</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>19/08/2024</td><td>24/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>35</td><td>Module02</td><td>STC CS SQL-217</td><td>
How to manage database security
1. How to work with SQL Server login IDs
a. An introduction to SQL Server security
b. How to change the authentication mode
c. How to create login IDs
d. How to change login IDs or passwords
e. How to work with database users
f. How to work with schemas
2. How to work with permissions
a. How to grant or revoke object permissions
b. The SQL Server object permissions
c. How to grant or revoke schema permissions
d. How to grant or revoke database permissions
e. How to grant or revoke server permissions
3. How to work with roles
a. How to work with the fixed server roles
b. How to work with user-defined server roles
c. How to display information about server roles and role members
d. How to work with the fixed database roles
e. How to work with user-defined database roles
f. How to display information about database roles and role members
g. How to deny permissions granted by role membership
h. How to work with application roles
4. How to manage security using the Management Studio
a. How to work with login IDs
b. How to work with the server roles for a login ID
c. How to assign database access and roles by login ID
d. How to assign user permissions to database objects
e. How to work with database permissions
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Work with SQL Server security, login IDs, passwords, users, schemas, permissions, roles and memberships</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>20/08/2024</td><td>25/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>36</td><td>Module03</td><td>STC CS PLCS-301</td><td>Welcome to C# and Microsoft Visual Studio 2019
1. Installing & Introducing Microsoft Visual Studio 2019
a. Installing Microsoft Visual Studio 2019
b. Introducing Microsoft Visual Studio 2019
c. Introducing Tools, Windows & Features of Microsoft Visual Studio 2019
2. Welcome to C#
a. Beginning programming with the Visual Studio 2019 environment
b. Writing your first program
c. Using namespaces
d. Creating a graphical application
i. Examining the Universal Windows Platform app
ii. Adding code to the graphical application

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Install Microsoft Visual Studio 2019
• Use the tools, windows and features of Microsoft Visual Studio 2019
• Understand namespace
• Create basic console and graphical application
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>21/08/2024</td><td>27/08/2024</td></tr>
<tr><td>37</td><td>Module03</td><td>STC CS PLCS-301</td><td>Working with variables, operators, and expressions
1. Understanding statements
2. Using identifiers
a. Identifying keywords
3. Using variables
a. Naming variables
b. Declaring variables
c. Specifying numeric values
4. Working with primitive data types
a. Unassigned local variables
b. Displaying primitive data type values
5. Using arithmetic operators
a. Operators and types
b. Examining arithmetic operators
c. Controlling precedence
d. Using associativity to evaluate expressions
e. Associativity and the assignment operator
6. Incrementing and decrementing variables
a. Prefix and postfix
7. Declaring implicitly typed local variables

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understand, declare, use and work with statements, identifiers, variables, primitive data types, arithmetic operators, increment & decrement variables, implicit type local variables
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>22/08/2024</td><td>28/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>38</td><td>Module03</td><td>STC CS PLCS-301</td><td>Working with variables, operators, and expressions
1. Understanding statements
2. Using identifiers
a. Identifying keywords
3. Using variables
a. Naming variables
b. Declaring variables
c. Specifying numeric values
4. Working with primitive data types
a. Unassigned local variables
b. Displaying primitive data type values
5. Using arithmetic operators
a. Operators and types
b. Examining arithmetic operators
c. Controlling precedence
d. Using associativity to evaluate expressions
e. Associativity and the assignment operator
6. Incrementing and decrementing variables
a. Prefix and postfix
7. Declaring implicitly typed local variables
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand, declare, use and work with statements, identifiers, variables, primitive data types, arithmetic operators, increment & decrement variables, implicit type local variables</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>24/08/2024</td><td>29/08/2024</td></tr>
<tr><td>39</td><td>Module03</td><td>STC CS PLCS-301</td><td>Writing methods and applying scope
1. Creating methods
a. Declaring a method
b. Returning data from a method
c. Using expression-bodied methods
d. Calling methods
e. Specifying the method call syntax
f. Returning multiple values from a method
2. Applying scope
a. Defining local scope
b. Defining class scope
c. Overloading methods
3. Writing methods
a. Refactoring code
b. Nesting methods
4. Using optional parameters and named arguments
a. Defining optional parameters
b. Passing named arguments
c. Resolving ambiguities with optional parameters and named arguments
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create method, apply scope, refactor code, nest method, use optional parameter and named argument</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>25/08/2024</td><td>31/08/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>40</td><td>Module03</td><td>STC CS PLCS-301</td><td>Writing methods and applying scope
1. Creating methods
a. Declaring a method
b. Returning data from a method
c. Using expression-bodied methods
d. Calling methods
e. Specifying the method call syntax
f. Returning multiple values from a method
2. Applying scope
a. Defining local scope
b. Defining class scope
c. Overloading methods
3. Writing methods
a. Refactoring code
b. Nesting methods
4. Using optional parameters and named arguments
a. Defining optional parameters
b. Passing named arguments
c. Resolving ambiguities with optional parameters and named arguments
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• create method, apply scope, refactor code, nest method, use optional parameter and named argument
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>27/08/2024</td><td>01/09/2024</td></tr>
<tr><td>41</td><td>Module03</td><td>STC CS PLCS-301</td><td>Using decision statements
1. Declaring Boolean variables
2. Using Boolean operators
a. Understanding equality and relational operators
b. Understanding conditional logical operators
c. Short-circuiting
d. Summarizing operator precedence and associativity
3. Using if statements to make decisions
a. Understanding if statement syntax
b. Using blocks to group statements
c. Cascading if statements
4. Using switch statements
a. Understanding switch statement syntax
b. Following the switch statement rules
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• declare Boolean variable, use Boolean operator
• use if and cascading if statements
• use switch statement and know switch statement rules
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.
a</td><td>28/08/2024</td><td>02/09/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>42</td><td>Module03</td><td>STC CS PLCS-301</td><td>Using compound assignment and iteration statements
1. Using compound assignment operators
2. Writing while statements
3. Writing for statements
a. Understanding for statement scope
4. Writing do statements
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• use compound assignment operators
• write while, for and do statements</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>29/08/2024</td><td>03/09/2024</td></tr>
<tr><td>43</td><td>Module03</td><td>STC CS PLCS-301</td><td>Managing errors and exceptions
1. Coping with errors
2. Trying code and catching exceptions
a. Unhandled exceptions
b. Using multiple catch handlers
c. Catching multiple exceptions
d. Filtering exceptions
e. Propagating exceptions
3. Using checked and unchecked integer arithmetic
a. Writing checked statements
b. Writing checked expressions
4. Throwing exceptions
5. Using throw exceptions
6. Using a finally block
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand errors
• understand and use try-catch-finally block
• throw exception</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>31/08/2024</td><td>04/09/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>44</td><td>Module03</td><td>STC CS PLCS-302</td><td>Creating and managing classes and objects
1. Understanding classification
2. The purpose of encapsulation
3. Defining and using a class
4. Controlling accessibility
a. Working with constructors
b. Overloading constructors
c. Deconstructing an object
5. Understanding static methods and data
a. Creating a shared field
b. Creating a static field by using the const keyword
c. Understanding static classes
d. Static using statements
6. Anonymous classes
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• understand classes, constructors and methods
• define and use classes, constructors and methods
• understand the purpose of encapsulation
• can overload constructor, deconstruct an object
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>01/09/2024</td><td>05/09/2024</td></tr>
<tr><td>45</td><td>Module03</td><td>STC CS PLCS-302</td><td>Creating and managing classes and objects
1. Understanding classification
2. The purpose of encapsulation
3. Defining and using a class
4. Controlling accessibility
a. Working with constructors
b. Overloading constructors
c. Deconstructing an object
5. Understanding static methods and data
a. Creating a shared field
b. Creating a static field by using the const keyword
c. Understanding static classes
d. Static using statements
6. Anonymous classes
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• understand classes, constructors and methods
• define and use classes, constructors and methods
• understand the purpose of encapsulation
• can overload constructor, deconstruct an object
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>02/09/2024</td><td>09/09/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>46</td><td>Module03</td><td>STC CS PLCS-302</td><td>Understanding values and references
1. Copying value type variables and classes
2. Understanding null values and nullable types
a. The null-conditional operator
b. Using nullable types
c. Understanding the properties of nullable types
3. Using ref and out parameters
a. Creating ref parameters
b. Creating out parameters
4. How computer memory is organized
a. Using the stack and the heap
5. The System.Object class
6. Boxing
7. Unboxing
8. Casting data safely
a. The is operator
b. The as operator
c. The switch statement revisited
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand values and references
• copy value type variables and classes
• understand and handle null values and nullable types
• use ref and out parameters
• use stack and heap
• understand System.Object class
• understand boxing and unboxing
• safe casting using is and as operator</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>03/09/2024</td><td>10/09/2024</td></tr>
<tr><td>47</td><td>Module03</td><td>STC CS PLCS-302</td><td>Creating value types with enumerations and structures
1. Working with enumerations
a. Declaring an enumeration
b. Using an enumeration
c. Choosing enumeration literal values
d. Choosing an enumeration’s underlying type
2. Working with structures
a. Declaring a structure
b. Understanding differences between structures and classes
c. Declaring structure variables
d. Understanding structure initialization
e. Copying structure variables
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create and use enum and structure</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>04/09/2024</td><td>12/09/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>48</td><td>Module03</td><td>STC CS PLCS-302</td><td>Using arrays
1. Declaring and creating an array
a. Declaring array variables
b. Creating an array instance
2. Populating and using an array
a. Creating an implicitly typed array
b. Accessing an individual array element
c. Iterating through an array
d. Passing arrays as parameters and return values for a method
3. Copying arrays
4. Using multidimensional arrays
a. Creating jagged arrays
5. Accessing arrays that contain value types
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• understand, declare, create, populate, use, copy and access different types of arrays
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>05/09/2024</td><td>14/09/2024</td></tr>
<tr><td>49</td><td>Module03</td><td>STC CS PLCS-302</td><td>Understanding parameter arrays
1. Overloading—a recap
2. Using array arguments
a. Declaring a params Array
b. Using params object[]
c. Using a params array
3. Comparing parameter arrays and optional parameters
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• understand, declare and use parameter arrays
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>07/09/2024</td><td>15/09/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Preparatory Leave</td><td>07/09/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>50</td><td>Module03</td><td>STC CS PLCS-302</td><td>Working with inheritance
1. What is inheritance?
2. Using inheritance
a. The System.Object class revisited
b. Calling base class constructors
c. Assigning classes
d. Declaring new methods
e. Declaring virtual methods
f. Declaring override methods
g. Understanding protected access
3. Creating extension methods
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand and use inheritance
• create extension methods</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>08/09/2024</td><td>17/09/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Exam</td><td>08/09/2024</td><td></td></tr>
<tr><td>51</td><td>Module03</td><td>STC CS PLCS-302</td><td>Working with inheritance
1. What is inheritance?
2. Using inheritance
a. The System.Object class revisited
b. Calling base class constructors
c. Assigning classes
d. Declaring new methods
e. Declaring virtual methods
f. Declaring override methods
g. Understanding protected access
3. Creating extension methods
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand and use inheritance
• create extension methods</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>09/09/2024</td><td>18/09/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>52</td><td>Module03</td><td>STC CS PLCS-302</td><td>Creating interfaces and defining abstract classes
1. Understanding interfaces
a. Defining an interface
b. Implementing an interface
c. Referencing a class through its interface
d. Working with multiple interfaces
e. Explicitly implementing an interface
f. Interface restrictions
g. Defining and using interfaces
2. Abstract classes
a. Abstract methods
3. Sealed classes
a. Sealed methods
b. Implementing and using an abstract class
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand, define, implement and work with multiple interfaces
• reference a class through an interface
• understand interface restrictions
• understand, create and implement abstract classes
• understand, create and implement sealed classes</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>10/09/2024</td><td>19/09/2024</td></tr>
<tr><td>53</td><td>Module03</td><td>STC CS PLCS-302</td><td>Creating interfaces and defining abstract classes
1. Understanding interfaces
a. Defining an interface
b. Implementing an interface
c. Referencing a class through its interface
d. Working with multiple interfaces
e. Explicitly implementing an interface
f. Interface restrictions
g. Defining and using interfaces
2. Abstract classes
a. Abstract methods
3. Sealed classes
a. Sealed methods
b. Implementing and using an abstract class
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand, define, implement and work with multiple interfaces
• reference a class through an interface
• understand interface restrictions
• understand, create and implement abstract classes
• understand, create and implement sealed classes</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>11/09/2024</td><td>21/09/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Faculty Sick</td><td>11/09/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>54</td><td>Module03</td><td>STC CS PLCS-302</td><td>Creating interfaces and defining abstract classes
1. Understanding interfaces
a. Defining an interface
b. Implementing an interface
c. Referencing a class through its interface
d. Working with multiple interfaces
e. Explicitly implementing an interface
f. Interface restrictions
g. Defining and using interfaces
2. Abstract classes
a. Abstract methods
3. Sealed classes
a. Sealed methods
b. Implementing and using an abstract class
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand, define, implement and work with multiple interfaces
• reference a class through an interface
• understand interface restrictions
• understand, create and implement abstract classes
• understand, create and implement sealed classes</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>12/09/2024</td><td>22/09/2024</td></tr>
<tr><td>55</td><td>Module03</td><td>STC CS PLCS-302</td><td>Using garbage collection and resource management
1. The life and times of an object
a. Writing destructors
b. Why use the garbage collector?
c. How does the garbage collector work?
d. Recommendations
2. Resource management
a. Disposal methods
b. Exception-safe disposal
c. The using statement and the IDisposable interface
d. Calling the Dispose method from a destructor
3. Implementing exception-safe disposal
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand the lifetime of an object
• write an destructor
• understand the necessity of garbage collector and how does it works
• understand, create and implement disposal method</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>14/09/2024</td><td>23/09/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>56</td><td>Module03</td><td>STC CS PLCS-303</td><td>Implementing properties to access fields
1. Implementing encapsulation by using methods
2. What are properties?
a. Using properties
b. Read-only properties
c. Write-only properties
d. Property accessibility
3. Understanding the property restrictions
4. Declaring interface properties
a. Replacing methods with properties
5. Generating automatic properties
6. Initializing objects by using properties
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create and use properties
• understand property restrictions
• declare interface properties
• replace methods with properties
• generate automate properties
• initialize objects by using properties</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>15/09/2024</td><td>24/09/2024</td></tr>
<tr><td>57</td><td>Module03</td><td>STC CS PLCS-303</td><td>Handling binary data and using indexers
1. What is an indexer?
a. Storing binary values
b. Displaying binary values
c. Manipulating binary values
d. Solving the same problems using indexers
e. Understanding indexer accessors
f. Comparing indexers and arrays
2. Indexers in interfaces
3. Using indexers in a Windows application
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand indexer and indexer accessors
• display and manipulate binary values
• compare indexers and arrays
• using indexers in interface and windows application</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>17/09/2024</td><td>25/09/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>58</td><td>Module03</td><td>STC CS PLCS-303</td><td>Introducing generics
1. The problem: Misusing with the object type
2. The generics solution
a. Generics vs. generalized classes
b. Generics and constraints
3. Creating a generic class
a. The theory of binary trees
b. Building a binary tree class by using generics
4. Creating a generic method
a. Defining a generic method to build a binary tree
5. Variance and generic interfaces
a. Covariant interfaces
b. Contravariant interfaces
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand generic approach of OOP
• understand generic solution
• create and use generic classes and generic methods
• understand, create and implement covariant and contravariant interfaces</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>18/09/2024</td><td>26/09/2024</td></tr>
<tr><td>59</td><td>Module03</td><td>STC CS PLCS-303</td><td>Introducing generics
1. The problem: Misusing with the object type
2. The generics solution
a. Generics vs. generalized classes
b. Generics and constraints
3. Creating a generic class
a. The theory of binary trees
b. Building a binary tree class by using generics
4. Creating a generic method
a. Defining a generic method to build a binary tree
5. Variance and generic interfaces
a. Covariant interfaces
b. Contravariant interfaces
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand generic approach of OOP
• understand generic solution
• create and use generic classes and generic methods
• understand, create and implement covariant and contravariant interfaces</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>19/09/2024</td><td>28/09/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>60</td><td>Module03</td><td>STC CS PLCS-303</td><td>Using collections
1. What are collection classes?
a. The List<t> collection class
b. The LinkedList<t> collection class
c. The Queue<t> collection class
d. The Stack<t> collection class
e. The Dictionary<tkey, tvalue=""> collection class
f. The SortedList<tkey, tvalue=""> collection class
g. The HashSet<t> collection class
2. Using collection initializers
a. The Find methods, predicates, and lambda expressions
b. The forms of lambda expressions
3. Comparing arrays and collections
a. Using collection classes to play cards
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• understand collection class
• use List<t>, LinkedList<t>, Queue<t>, Stack<t>, Dictionary<tkey, tvalue="">, SortedList<tkey, tvalue="">, HashSet<t> collection classes
• understand and use lambda expressions
• compare arrays and collections
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>21/09/2024</td><td>29/09/2024</td></tr>
<tr><td>61</td><td>Module03</td><td>STC CS PLCS-303</td><td>Using collections
1. What are collection classes?
a. The List<t> collection class
b. The LinkedList<t> collection class
c. The Queue<t> collection class
d. The Stack<t> collection class
e. The Dictionary<tkey, tvalue=""> collection class
f. The SortedList<tkey, tvalue=""> collection class
g. The HashSet<t> collection class
2. Using collection initializers
a. The Find methods, predicates, and lambda expressions
b. The forms of lambda expressions
3. Comparing arrays and collections
a. Using collection classes to play cards
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• understand collection class
• use List<t>, LinkedList<t>, Queue<t>, Stack<t>, Dictionary<tkey, tvalue="">, SortedList<tkey, tvalue="">, HashSet<t> collection classes
• understand and use lambda expressions
• compare arrays and collections
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>22/09/2024</td><td>30/09/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>62</td><td>Module03</td><td>STC CS PLCS-303</td><td>Enumerating collections
1. Enumerating the elements in a collection
a. Manually implementing an enumerator
b. Implementing the IEnumerable interface
2. Implementing an enumerator by using an iterator
a. A simple iterator
b. Defining an enumerator for the Tree<TItem> class by using an iterator
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand and use enumerator
• understand and use iterator
• understand and define Tree<TItem> class by using an iterator</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>23/09/2024</td><td>01/10/2024</td></tr>
<tr><td>63</td><td>Module03</td><td>STC CS PLCS-303</td><td>Decoupling application logic and handling events
1. Understanding delegates
a. Examples of delegates in the .NET Framework class library
b. The automated factory scenario
c. Implementing the factory control system without using delegates
d. Implementing the factory by using a delegate
e. Declaring and using delegates
2. Lambda expressions and delegates
a. Creating a method adapter
3. Enabling notifications by using events
a. Declaring an event
b. Subscribing to an event
c. Unsubscribing from an event
d. Raising an event
4. Understanding user interface events
a. Using events
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand and use delegates, lambda expressions and events</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>24/09/2024</td><td>02/10/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>64</td><td>Module03</td><td>STC CS PLCS-303</td><td>Decoupling application logic and handling events
1. Understanding delegates
a. Examples of delegates in the .NET Framework class library
b. The automated factory scenario
c. Implementing the factory control system without using delegates
d. Implementing the factory by using a delegate
e. Declaring and using delegates
2. Lambda expressions and delegates
a. Creating a method adapter
3. Enabling notifications by using events
a. Declaring an event
b. Subscribing to an event
c. Unsubscribing from an event
d. Raising an event
4. Understanding user interface events
a. Using events
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand and use delegates, lambda expressions and events</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>25/09/2024</td><td>03/10/2024</td></tr>
<tr><td>65</td><td>Module03</td><td>STC CS PLCS-303</td><td>Querying in-memory data by using query expressions
1. What is LINQ?
2. Using LINQ in a C# application
a. Selecting data
b. Filtering data
c. Ordering, grouping, and aggregating data
d. Joining data
e. Using query operators
f. Querying data in Tree objects
g. LINQ and deferred evaluation
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand and use LINQ</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>26/09/2024</td><td>05/10/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>66</td><td>Module03</td><td>STC CS PLCS-303</td><td>Operator overloading
1. Understanding operators
a. Operator constraints
b. Overloaded operators
c. Creating symmetric operators
2. Understanding compound assignment evaluation
3. Declaring increment and decrement operators
4. Comparing operators in structures and classes
5. Defining operator pairs
6. Implementing operators
7. Understanding conversion operators
a. Providing built-in conversions
b. Implementing user-defined conversion operators
c. Creating symmetric operators, revisited
d. Writing conversion operators

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• understand and use operators, operators constraints, overload operators, create symmetric operators, conversion operators, implement user-defined conversion operators
• declare increment and decrement operator
• compare operators in structures and classes
• define operator pairs
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>28/09/2024</td><td>06/10/2024</td></tr>
<tr><td>67</td><td>Module03</td><td>STC CS PLCS-304</td><td>Improving throughput by using tasks
1. Why perform multitasking by using parallel processing?
a. The rise of the multicore processor
2. Implementing multitasking by using the Microsoft .NET Framework
a. Tasks, threads, and the ThreadPool
b. Creating, running, and controlling tasks
c. Using the Task class to implement parallelism
d. Abstracting tasks by using the Parallel class
e. When not to use the Parallel class
3. Canceling tasks and handling exceptions
a. The mechanics of cooperative cancellation
b. Using continuations with canceled and faulted tasks

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• understand and implement multitask, thread and the ThreadPool
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>29/09/2024</td><td>07/10/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>68</td><td>Module03</td><td>STC CS PLCS-304</td><td>Improving throughput by using tasks
1. Why perform multitasking by using parallel processing?
a. The rise of the multicore processor
2. Implementing multitasking by using the Microsoft .NET Framework
a. Tasks, threads, and the ThreadPool
b. Creating, running, and controlling tasks
c. Using the Task class to implement parallelism
d. Abstracting tasks by using the Parallel class
e. When not to use the Parallel class
3. Canceling tasks and handling exceptions
a. The mechanics of cooperative cancellation
b. Using continuations with canceled and faulted tasks
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand and implement multitask, thread and the ThreadPool</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>30/09/2024</td><td>12/10/2024</td></tr>
<tr><td>69</td><td>Module03</td><td>STC CS PLCS-304</td><td>Improving response time by performing asynchronous operations
1. Implementing asynchronous methods
a. Defining asynchronous methods: The problem
b. Defining asynchronous methods: The solution
c. Defining asynchronous methods that return values
d. Asynchronous method gotchas
e. Asynchronous methods and the Windows Runtime APIs
f. Tasks, memory allocation, and efficiency
2. Using PLINQ to parallelize declarative data access
a. Using PLINQ to improve performance while iterating through a collection
b. Canceling a PLINQ query
3. Synchronizing concurrent access to data
a. Locking data
b. Synchronization primitives for coordinating tasks
c. Canceling synchronization
d. The concurrent collection classes
e. Using a concurrent collection and a lock to implement thread-safe data access
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand and implement asynchronous operation</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>01/10/2024</td><td>14/10/2024</td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
70	Module04	STC CS MSSL-401	**Configure Development Environment, HTML Elements, Styles and Semantic HTML** 1. SETTING UP A DEVELOPMENT ENVIRONMENT a. Install a Code Editor i. Install Essential Extensions ii. Install Live Server iii. Setting Default Formatter iv. Enable Formatting Code on Save v. Auto Save Files on Focus Change vi. Disable Compact Folders vii. Enable Word Wrap viii. Set Tab Size Smaller b. Using a Web Browser i. Using Developer Tools in The Browser ii. Using Console Tool iii. Using Source Tool c. Preparing a Workspace 2. GETTING TO KNOW HTML a. Two Types of HTML Elements b. HTML Document Structure c. Your First Webpage d. Character Encoding e. Element Attributes f. Favorite Icon g. Comments 3. HTML ELEMENTS a. Headings b. Paragraphs c. Hyperlinks i. External Anchors ii. Internal Anchors iii. Download Links d. Images i. Image Maps ii. Responsive Pictures e. Lists i. Unordered List ii. Ordered List iii. Description Lists f. Tables i. Table Headers ii. Table Rows iii. Table Caption iv. Group of Columns v. Column Span g. Breaks i. Line Breaking ii. Thematic Breaking h. Progress Bars i. progress ii. meter i. Computer Code i. code ii. pre j. Iframes k. HTML Entities 4. HTML STYLES a. Formatting Elements i. b ii. i iii. u or ins iv. s or del v. mark vi. sub and sup b. HTML Style Attribute 5. SEMANTIC HTML a. Semantic Elements i. blockquote ii. address iii. details and summary iv. figure and figcaption b. Semantic Layout Elements c. Semantic Formatting Elements	02/10/2024	15/10/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>i. strong
ii. em
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Setup HTML development environment.
understand and use HTML
use HTML Elements
use HTML Styles
use Semantic HTML</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td></td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
71	Module04	STC CS MSSL-401	**Configure Development Environment, HTML Elements, Styles and Semantic HTML** 1. SETTING UP A DEVELOPMENT ENVIRONMENT a. Install a Code Editor i. Install Essential Extensions ii. Install Live Server iii. Setting Default Formatter iv. Enable Formatting Code on Save v. Auto Save Files on Focus Change vi. Disable Compact Folders vii. Enable Word Wrap viii. Set Tab Size Smaller b. Using a Web Browser i. Using Developer Tools in The Browser ii. Using Console Tool iii. Using Source Tool c. Preparing a Workspace 2. GETTING TO KNOW HTML a. Two Types of HTML Elements b. HTML Document Structure c. Your First Webpage d. Character Encoding e. Element Attributes f. Favorite Icon g. Comments 3. HTML ELEMENTS a. Headings b. Paragraphs c. Hyperlinks i. External Anchors ii. Internal Anchors iii. Download Links d. Images i. Image Maps ii. Responsive Pictures e. Lists i. Unordered List ii. Ordered List iii. Description Lists f. Tables i. Table Headers ii. Table Rows iii. Table Caption iv. Group of Columns v. Column Span g. Breaks i. Line Breaking ii. Thematic Breaking h. Progress Bars i. progress ii. meter i. Computer Code i. code ii. pre j. Iframes k. HTML Entities 4. HTML STYLES a. Formatting Elements i. b ii. i iii. u or ins iv. s or del v. mark vi. sub and sup b. HTML Style Attribute 5. SEMANTIC HTML a. Semantic Elements i. blockquote ii. address iii. details and summary iv. figure and figcaption b. Semantic Layout Elements c. Semantic Formatting Elements	03/10/2024	16/10/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>i. strong
ii. em
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Setup HTML development environment.
understand and use HTML
use HTML Elements
use HTML Styles
use Semantic HTML</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td></td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
72	Module04	STC CS MSSL-401	**Configure Development Environment, HTML Elements, Styles and Semantic HTML** 1. SETTING UP A DEVELOPMENT ENVIRONMENT a. Install a Code Editor i. Install Essential Extensions ii. Install Live Server iii. Setting Default Formatter iv. Enable Formatting Code on Save v. Auto Save Files on Focus Change vi. Disable Compact Folders vii. Enable Word Wrap viii. Set Tab Size Smaller b. Using a Web Browser i. Using Developer Tools in The Browser ii. Using Console Tool iii. Using Source Tool c. Preparing a Workspace 2. GETTING TO KNOW HTML a. Two Types of HTML Elements b. HTML Document Structure c. Your First Webpage d. Character Encoding e. Element Attributes f. Favorite Icon g. Comments 3. HTML ELEMENTS a. Headings b. Paragraphs c. Hyperlinks i. External Anchors ii. Internal Anchors iii. Download Links d. Images i. Image Maps ii. Responsive Pictures e. Lists i. Unordered List ii. Ordered List iii. Description Lists f. Tables i. Table Headers ii. Table Rows iii. Table Caption iv. Group of Columns v. Column Span g. Breaks i. Line Breaking ii. Thematic Breaking h. Progress Bars i. progress ii. meter i. Computer Code i. code ii. pre j. Iframes k. HTML Entities 4. HTML STYLES a. Formatting Elements i. b ii. i iii. u or ins iv. s or del v. mark vi. sub and sup b. HTML Style Attribute 5. SEMANTIC HTML a. Semantic Elements i. blockquote ii. address iii. details and summary iv. figure and figcaption b. Semantic Layout Elements c. Semantic Formatting Elements	05/10/2024	17/10/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>i. strong
ii. em
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Setup HTML development environment.
understand and use HTML
use HTML Elements
use HTML Styles
use Semantic HTML</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td></td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>73</td><td>Module04</td><td>STC CS MSSL-402</td><td>
WEB FORMS
1. HTML Form Elements
a. input and label
b. textarea
c. select and option
d. button
e. fieldset and legend
f. datalist
2. HTML Input Types
a. Buttons
i. button
ii. submit
iii. reset
iv. image
b. Texts
i. text
ii. password
iii. email
iv. url
v. hidden
c. Numbers
i. number
ii. range
d. Options
i. checkbox
ii. radio
e. Files
f. Date and Time
i. date
ii. time
iii. datetime-local
g. Colors
3. HTML Input Attributes
a. name
b. value
c. placeholder
d. readonly
e. disabled
f. size
g. multiple
h. step
i. width and height
j. autofocus
k. autocomplete
4. Form Validation
a. minlength
b. maxlength
c. min and max
d. required
e. pattern
f. Styles for The Invalid Inputs
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Use HTML Form and It's elements
• Use all HTML input types
• Use HTML input attributes
• Validate HTML form using form validation</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>06/10/2024</td><td>19/10/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>74</td>
<td>Module04</td>
<td>STC CS MSSL-402</td>
<td>
WEB FORMS

HTML Form Elements
<ol type="a">
input and label
textarea
select and option
button
fieldset and legend
datalist

HTML Input Types
<ol type="a">
Buttons
<ol type="i">
button
submit
reset
image

Texts
<ol type="i">
text
password
email
url
hidden

Numbers
<ol type="i">
number
range

Options
<ol type="i">
checkbox
radio

Files
Date and Time
<ol type="i">
date
time
datetime-local

Colors

HTML Input Attributes
<ol type="a">
name
value
placeholder
readonly
disabled
size
multiple
step
width and height
autofocus
autocomplete

Form Validation
<ol type="a">
minlength
maxlength
min and max
required
pattern
Styles for The Invalid Inputs

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Use HTML Form and It's elements
Use all HTML input types
Use HTML input attributes
Validate HTML form using form validation</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>07/10/2024</td>
<td>20/10/2024</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>75</td><td>Module04</td><td>STC CS MSSL-403</td><td>
HTML MULTIMEDIA, GRAPHICS & ADVANCE HTML
<ol type="1">
HTML MULTIMEDIA
<ol type="a">
Audio
Video

HTML GRAPHICS
<ol type="a">
SVG Graphics
<ol type="i">
SVG Rectangle
SVG Circle
SVG Ellipse
SVG Line
SVG Polyline
SVG Polygon
SVG Path
SVG Text
SVG Link
SVG Stroke

Stroke Color
Stroke Width
Stroke Ending
Dash Stroke

SVG Gradients

Linear Gradient
Radial Gradient

SVG Filters

Canvas
<ol type="i">
Draw Text
Draw Lines
Draw Circles
Draw Rectangles
Draw Gradients

Linear Gradient
Radial Gradient

Draw Images

HTML ADVANCED
<ol type="a">
Block Elements
Inline Elements
Inline-block Elements

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Use HTML multimedia, graphics
Work with Advanced HTML</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>08/10/2024</td><td>21/10/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>External Exam</td><td>08/10/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>76</td><td>Module04</td><td>STC CS MSSL-403</td><td>
HTML MULTIMEDIA, GRAPHICS & ADVANCE HTML
<ol type="1">
HTML MULTIMEDIA
<ol type="a">
Audio
Video

HTML GRAPHICS
<ol type="a">
SVG Graphics
<ol type="i">
SVG Rectangle
SVG Circle
SVG Ellipse
SVG Line
SVG Polyline
SVG Polygon
SVG Path
SVG Text
SVG Link
SVG Stroke

Stroke Color
Stroke Width
Stroke Ending
Dash Stroke

SVG Gradients

Linear Gradient
Radial Gradient

SVG Filters

Canvas
<ol type="i">
Draw Text
Draw Lines
Draw Circles
Draw Rectangles
Draw Gradients

Linear Gradient
Radial Gradient

Draw Images

HTML ADVANCED
<ol type="a">
Block Elements
Inline Elements
Inline-block Elements

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Use HTML multimedia, graphics
Work with Advanced HTML</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>09/10/2024</td><td>22/10/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>exam</td><td>09/10/2024</td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
77	Module04	STC CS MSSL-404	**GETTING TO KNOW CSS and CSS SELECTORS** 1. GETTING TO KNOW CSS a. CSS Syntax b. Comments in CSS c. CSS Units i. Absolute Lengths ii. Relative Lengths ▪ em ▪ rem iii. Shorthand Properties iv. Where Do We Put Our CSS? i. Inline CSS ii. Internal CSS iii. External CSS v. User Agent Style Sheets 2. CSS SELECTORS a. Universal Selector b. Type Selector c. ID Selector d. Class Selector e. Multiple-class Selector f. CSS Pseudo-class i. Status of Links ▪ :link ▪ :visited ii. Cursor Interaction ▪ :focus ▪ :hover ▪ :active iii. Status of Elements ▪ :enabled ▪ :disabled ▪ :checked iv. Form Validation ▪ :invalid ▪ :required ▪ :out-of-range v. Child Elements ▪ :first-child ▪ :last-child ▪ :nth-child(N) ▪ :nth-last-child(N) ▪ :only-child vi. Type of elements ▪ :first-of-type ▪ :last-of-type ▪ :nth-of-type(N) ▪ :nth-last-of-type(N) ▪ :only-of-type vii. Others Pseudo-classes ▪ :root ▪ :empty ▪ :target ▪ :not(selector) viii. Pseudo-class Chain g. CSS Pseudo-element i. ::first-letter ii. ::first-line iii. ::before iv. ::after v. ::selection h. CSS Combined Selector i. Descendant selector ii. Child Selector iii. Adjacent Sibling Selector iv. General Sibling Selector i. Attribute Selector i. [attribute] ii. [attribute="value"] iii. [attribute|="value"] iv. [attribute~="value"] v. [attribute^="value"] vi. [attribute$="value"]	12/10/2024	23/10/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>vii. [attribute*="value"]
j. Selector Grouping
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Learn types and why & how to use CSS selectors
Learn why & how to use CSS Pseudo-element
Learn why & how to use CSS combined selector
Learn why & how o use CSS attribute selector</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td></td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
78	Module04	STC CS MSSL-404	**GETTING TO KNOW CSS and CSS SELECTORS** 1. GETTING TO KNOW CSS a. CSS Syntax b. Comments in CSS c. CSS Units i. Absolute Lengths ii. Relative Lengths ▪ em ▪ rem iii. Shorthand Properties iv. Where Do We Put Our CSS? i. Inline CSS ii. Internal CSS iii. External CSS v. User Agent Style Sheets 2. CSS SELECTORS a. Universal Selector b. Type Selector c. ID Selector d. Class Selector e. Multiple-class Selector f. CSS Pseudo-class i. Status of Links ▪ :link ▪ :visited ii. Cursor Interaction ▪ :focus ▪ :hover ▪ :active iii. Status of Elements ▪ :enabled ▪ :disabled ▪ :checked iv. Form Validation ▪ :invalid ▪ :required ▪ :out-of-range v. Child Elements ▪ :first-child ▪ :last-child ▪ :nth-child(N) ▪ :nth-last-child(N) ▪ :only-child vi. Type of elements ▪ :first-of-type ▪ :last-of-type ▪ :nth-of-type(N) ▪ :nth-last-of-type(N) ▪ :only-of-type vii. Others Pseudo-classes ▪ :root ▪ :empty ▪ :target ▪ :not(selector) viii. Pseudo-class Chain g. CSS Pseudo-element i. ::first-letter ii. ::first-line iii. ::before iv. ::after v. ::selection h. CSS Combined Selector i. Descendant selector ii. Child Selector iii. Adjacent Sibling Selector iv. General Sibling Selector i. Attribute Selector i. [attribute] ii. [attribute="value"] iii. [attribute|="value"] iv. [attribute~="value"] v. [attribute^="value"] vi. [attribute$="value"]	14/10/2024	24/10/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>vii. [attribute*="value"]
j. Selector Grouping
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Learn types and why & how to use CSS selectors
Learn why & how to use CSS Pseudo-element
Learn why & how to use CSS combined selector
Learn why & how o use CSS attribute selector</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td></td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
79	Module04	STC CS MSSL-405	**CSS CASCADING, BOX MODEL, COLORS, TEXT, BACKGROUNDS** 1. CSS CASCADING a. Rule 1: The Last Declaration Will Win b. Rule 2: The More Specific Declaration Will Win c. Rule 3: Inline Style Is Preferred to both Internal and External Styles d. Rule 4: "!important" Declaration Is the Strongest Rule e. Inheritance 2. CSS BOX MODEL a. Box Sizing Types i. Border Box ii. Content Box b. Margins i. Auto Margin ii. Collapsing Margins c. CSS Outline d. CSS Borders i. Border Styles ii. CSS Border Radius iii. CSS Border Images e. CSS Padding f. CSS Width and Height 3. CSS COLORS a. Color Names b. HEX Colors c. RGB Colors d. Grey Colors 4. CSS TEXT a. Text Rendering i. Setting Color ii. Text Decoration iii. Text Transform iv. Handling Whitespaces v. Text Overflow vi. Word Wrap vii. Word Break b. Text Spacing i. Letter Spacing ii. Word Spacing iii. Line Height c. Text Position i. Text Align ii. Text Indent d. CSS Fonts i. Font Family ii. Font Size iii. Font Weight iv. Font Style v. Font Variant vi. The @font-face 5. CSS BACKGROUNDS a. CSS Background Properties i. background-color ii. background-image iii. background-repeat iv. background-position v. background-size vi. background vii. background-clip viii. background-origin ix. background-attachment b. CSS Gradient Backgrounds i. Linear Gradients ii. Repeating Linear Gradients iii. Radial Gradients iv. Repeating Radial Gradients c. CSS Sprites **EVIDENCE GUIDANCE**	16/10/2024	26/10/2024

Sl	Module	Competency Unit	Elements		Schedule Date	Actual Date
			Critical aspects of evidence	**Assessment must confirm the ability to** – • Learn and implement CSS cascading roles • Learn and implement CSS box model, colors, text, backgrounds		
			Assessment of unit	This unit may be assessed by taking **minimum one (1) quiz** after completion of the unit. Each quiz will contain **10 questions of 1 mark each**. A **total mark of each quiz is 10**. **Quiz time is 10 minutes**.		

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
80	Module04	STC CS MSSL-406	**CSS DISPLAY & CSS LAYOUT** 1. CSS DISPLAY a. display vs. visibility i. display ii. visibility b. CSS object-fit c. Cursor d. CSS Overflow i. overflow: visible ii. overflow: hidden iii. overflow: scroll iv. overflow: auto e. CSS Opacity f. CSS Shadows i. text-shadow ii. box-shadow g. CSS Multiple Columns i. Fixed Columns ii. Responsive Columns iii. Column Span h. CSS Tables i. Table Sizing ii. Text Alignment in Table Cells i. CSS Lists i. List Item Markers ii. Position the List Item Markers j. CSS Transforms i. Moves an Element ii. Scales an Element iii. Rotates an Element iv. Skews an Element 2. CSS LAYOUT a. CSS Position i. static ii. relative iii. absolute iv. fixed v. sticky b. CSS Float i. float ii. clear c. CSS Z-index d. CSS Flexbox i. Flex Container ▪ gap ▪ flex-direction ▪ justify-content ▪ align-items ▪ flex-wrap ▪ align-content ii. Flex Items i. flex-basis ii. order iii. flex-grow iv. flex-shrink v. align-self vi. flex vii. Auto Margin for Flex Items e. CSS Grid i. Grid Container ▪ grid-template-columns ▪ grid-template-rows ▪ column-gap ▪ row-gap ▪ gap ▪ justify-items ▪ align-items ▪ justify-content ▪ align-content ii. Grid Items ▪ grid-column-start ▪ grid-column-end ▪ grid-column	17/10/2024	27/10/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>
grid-row-start
grid-row-end
grid-row
grid-area
justify-self
align-self
grid-template-areas
f. CSS Alignment
i. Horizontal Alignment
ii. Vertical alignment
iii. Perfect Centering
Centering Using The position And transform Properties
Centering Using The flex Property
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Learn and implement CSS display
Learn and implement CSS layout</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td></td><td></td></tr>
<tr><td>81</td><td>Module04</td><td>STC CS MSSL-407</td><td>
CSS TRANSITIONS AND ANIMATIONS, ADVANCED & RESPONSIVE CSS
1. CSS TRANSITIONS AND ANIMATIONS
a. CSS Transitions
i. Basic Transitions
ii. Transition Delay
iii. Transition Timing Function
iv. Shorthand Transition Property
b. CSS Animations
i. Two-step Keyframes
ii. Multiple Keyframes
iii. Animation Delay
iv. Animation Iteration Count
v. Animation Direction
vi. Animation Timing Function
vii. Shorthand Animation Property
2. ADVANCED CSS
a. CSS Variables
b. CSS Calculation
c. CSS Counters
3. RESPONSIVE CSS
a. Responsive Web Design
b. RWD Viewport
c. CSS Media
i. Media Types
ii. Media Queries
d. RWD Grid View
e. Mobile-first vs. Desktop-first
i. Mobile-First
ii. Desktop-First
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Learn and implement CSS transformations & animations
Learn and implement CSS variables, calculation & counters
Learn and implement responsive CSS</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>19/10/2024</td><td>28/10/2024</td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
82	Module04	STC CS MSSL-408	**The JavaScript Language** 1. Setting Up a Development Environment a. Choosing a Code Editor b. Using the Developer Console in Your Browser c. Running Blocks of Code in the Developer Console d. Using Strict Mode to Catch Common Mistakes e. Filling in HTML Boilerplate with Emmet Shortcuts f. Installing the npm Package Manager (with Node.js) g. Downloading a Package with npm h. Updating a Package with npm i. Setting Up a Local Test Server j. Enforcing Code Standards with a Linter k. Styling Code Consistently with a Formatter l. Experimenting in a JavaScript Playground 2. Strings and Regular Expressions a. Checking for an Existing, Nonempty String b. Converting a Numeric Value to a Formatted String c. Inserting Special Characters d. Inserting Emojis e. Using Template Literals for Clearer String Concatenation f. Performing a Case-Insensitive String Comparison g. Checking If a String Contains a Specific Substring h. Replacing All Occurrences of a String i. Replacing HTML Tags with Named Entities j. Using a Regular Expression to Replace Patterns in a String k. Extracting a List from a String l. Finding All Instances of a Pattern m. Removing Whitespace from the Beginning and End of a String n. Converting the First Letter of a String to Uppercase o. Validating an Email Address 3. Numbers a. Generating Random Numbers b. Generating Cryptographically Secure Random Numbers c. Rounding to a Specific Decimal Place d. Preserving Accuracy in Decimal Values e. Converting a String to a Number f. Converting a Decimal to a Hexadecimal Value g. Converting Between Degrees and Radians h. Calculating the Length of a Circular Arc i. Manipulating Very Large Numbers with BigInt 4. Dates a. Getting the Current Date and Time b. Converting a String to a Date c. Adding Days to a Date d. Comparing Dates and Testing Dates for Equality e. Calculating the Time Elapsed Between Two Dates f. Formatting a Date Value as a String 5. Arrays a. Checking If an Object Is an Array b. Iterating Over All the Elements in an Array c. Checking If Two Arrays Are Equal d. Breaking Down an Array into Separate Variables e. Passing an Array to a Function That Expects a List of Values f. Cloning an Array g. Merging Two Arrays h. Copying a Portion of an Array by Position i. Extracting Array Items That Meet Specific Criteria j. Emptying an Array k. Removing Duplicate Values l. Flattening a Two-Dimensional Array m. Searching Through an Array for Exact Matches n. Searching Through an Array for Items That Meet Specific Criteria o. Removing or Replacing Array Elements 107 p. Sorting an Array of Objects by a Property Value q. Transforming Every Element of an Array r. Combining an Array's Values in a Single Calculation s. Validating Array Contents t. Creating a Collection of Nonduplicated Values u. Creating a Key-Indexed Collection of Items **EVIDENCE GUIDANCE**	20/10/2024	29/10/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Learn setting up JavaScript development environment
Learn JavaScript strings and regular expressions
Learn JavaScript numbers, dates and arrays</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td></td><td></td></tr>
<tr><td>83</td><td>Module04</td><td>STC CS MSSL-408</td><td>
The JavaScript Language
6. Functions
a. Passing a Function as an Argument to Another Function
b. Using Arrow Functions
c. Providing a Default Parameter Value
d. Creating a Function That Accepts Unlimited Arguments
e. Using Named Function Parameters
f. Creating a Function That Stores its State with a Closure
g. Creating a Generator Function That Yields Multiple Values
h. Reducing Redundancy by Using Partial Application
i. Fixing this with Function Binding
j. Implementing a Recursive Algorithm
7. Objects
a. Checking if an Object Is a Certain Type
b. Using an Object Literal to Bundle Data
c. Checking If an Object Has a Property
d. Iterating Over All the Properties of an Object
e. Testing for an Empty Object
f. Merging the Properties of Two Objects
g. Customizing the Way a Property Is Defined
h. Preventing Any Changes to an Object
i. Intercepting and Changing Actions on an Object with a Proxy
j. Cloning an Object
k. Making a Deep Copy of an Object
l. Creating Absolutely Unique Object Property Keys
m. Creating Enums with Symbol
8. Classes
a. Creating a Reusable Class
b. Adding Properties to a Class
c. Giving a Class a Better String Representation
182
d. Using the Constructor Pattern to Make a Custom Class
e. Supporting Method Chaining in Your Class
f. Adding Static Methods to a Class
g. Using a Static Method to Create Objects
h. Inheriting Functionality from Another Class
i. Organizing Your JavaScript Classes with Modules
<table>
<tr><td colspan="2">EVIDENCE GUIDANCE</td></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Learn JavaScript functions, objects and classes</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>21/10/2024</td><td>30/10/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>84</td><td>Module04</td><td>STC CS MSSL-408</td><td>
The JavaScript Language
9. Asynchronous Programming
a. Updating the Page During a Loop
b. Using a Function That Returns a Promise
c. Promisifying an Asynchronous Function That Uses a Callback
d. Executing Multiple Promises Concurrently
e. Waiting for a Promise to Finish with Await and Async
f. Creating an Asynchronous Generator Function
g. Using a Web Worker to Perform a Background Task
h. Adding Progress Support to a Web Worker
10. Errors and Testing
a. Catching and Neutralizing an Error
b. Catching Different Types of Errors
c. Catching Asynchronous Errors
d. Detecting Unhandled Errors
e. Throwing a Standard Error
f. Throwing a Custom Error
g. Writing Unit Tests for Your Code
h. Tracking Test Code Coverage
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Learn JavaScript asynchronous programming
• Learn error handling and testing in JavaScript</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>22/10/2024</td><td>31/10/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>85</td><td>Module04</td><td>STC CS MSSL-409</td><td>
JavaScript in the Browser
1. Browser Tools
a. Debugging JavaScript
b. Analyzing Runtime Performance
c. Identifying Unused JavaScript
d. Using Lighthouse to Measure Best Practices
2. Working with HTML
a. Accessing a Given Element and Finding Its Parent and Child Elements
b. Traversing the Results from querySelectorAll() with forEach()
c. Adding Click Functionality to an Element
d. Finding All Elements That Share an Attribute
e. Accessing All Elements of a Specific Type
f. Discovering Child Elements Using the Selectors API
g. Changing an Element's Class Value
h. Setting an Element's Style Attribute
i. Adding Text to a New Paragraph
j. Inserting a New Element in a Specific DOM Location
k. Checking If a Checkbox Is Checked
l. Adding Up Values in an HTML Table
m. Deleting Rows from an HTML Table
n. Hiding Page Sections
o. Creating Hover-Based Pop-Up Info Windows
p. Validating Form Data
q. Highlighting Form Errors and Accessibility
r. Creating an Accessible Automatically Updated Region
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Learn using browser tools
• Learn manipulating, changing, hiding & showing DOM using JavaScript
• Learn adding, removing, changing, hiding & showing HTML element using JavaScript
• Learn adding & changing element's attribute and value using JavaScript
• Learn to managing classes and styles using JavaScript
• Learn how to validate a form using JavaScript
• Implement basic UX using JavaScript</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>23/10/2024</td><td>02/11/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>86</td><td>Module04</td><td>STC CS MSSL-409</td><td>JavaScript in the Browser
3. Fetching Remote Data
a. Requesting Remote Data with Fetch
b. Using XMLHttpRequest
c. Submitting a Form
d. Populating a Selection List from the Server
e. Parsing Returned JSON
f. Fetching and Parsing XML
g. Sending Binary Data and Loading into an Image
h. Sharing HTTP Cookies Across Domains
i. Using Websockets to Establish a Two-Way Communication Between Client and Server
j. Long Polling a Remote Data Source
4. Data Persistence
a. Persisting Information with Cookies
b. Using sessionStorage for Client-Side Storage
c. Creating a localStorage Client-Side Data Storage Item
d. Persisting Larger Chunks of Data on the Client Using IndexedDB
e. Simplifying IndexedDB with a Library
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Learn fetching remote data using JavaScript
• Learn data persistency using JavaScript
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>24/10/2024</td><td>03/11/2024</td></tr>
<tr><td>87</td><td>Module04</td><td>STC CS MSSL-409</td><td>JavaScript in the Browser
5. Working with Media
a. Adding JavaScript to SVG
b. Accessing SVG from a Web Page Script
c. Creating an SVG Bar Chart with D3
d. Integrating SVG and the Canvas Element in HTML
e. Running a Routine When an Audio File Begins Playing
f. Controlling Video from JavaScript with the video Element
6. Writing Web Applications
a. Bundling JavaScript
b. JavaScript and the Mobile Web
c. Writing a Progressive Web Application
d. Testing and Profiling a Progressive Web Application
e. Getting the Value of the Current URL
f. Redirecting a URL
g. Copying Text to a User's Clipboard
h. Enabling a Mobile-Like Notification in the Desktop Browser
i. Loading a File Locally in the Browser
j. Extending the Possible with Web Components
k. Choosing a Front-End Framework
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Learn working with media using JavaScript
• Learn to write web applications using JavaScript
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>26/10/2024</td><td>04/11/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>88</td><td>Module04</td><td>STC CS MSSL-410</td><td>
DATA URIs & HTML EVENTS and HTML APIs

Data URIs
Window Events
<ol type="a">
onload
onresize

Form Events
<ol type="a">
onfocus
onblur
onchange
oninput

Keyboard Events
<ol type="a">
onkeydown
onkeyup
onkeypress

Mouse Events
<ol type="a">
onclick
ondblclick
onmousemove
onmousedown and onmouseup
onmouseover and onmouseout

Clipboard Events
<ol type="a">
oncopy
oncut
onpaste

HTML APIs
<ol type="a">
Drag and Drop
Geolocation
Web Storage
<ol type="i">
The localStorage Object
The sessionStorage Object

Web Workers
<ol type="i">
Web Worker File
Web Worker Object
Web Workers and the DOM

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –

use data URIs
use window events
use form events
use keyboard events
use mouse events
use clipboard events
use drag and drop
use geolocation
use web storage
use web workers
</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>27/10/2024</td><td>06/11/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>89</td><td>Module04</td><td>STC CS MSSL-410</td><td>
DATA URIs & HTML EVENTS and HTML APIs

Data URIs
Window Events
<ol type="a">
onload
onresize

Form Events
<ol type="a">
onfocus
onblur
onchange
oninput

Keyboard Events
<ol type="a">
onkeydown
onkeyup
onkeypress

Mouse Events
<ol type="a">
onclick
ondblclick
onmousemove
onmousedown and onmouseup
onmouseover and onmouseout

Clipboard Events
<ol type="a">
oncopy
oncut
onpaste

HTML APIs
<ol type="a">
Drag and Drop
Geolocation
Web Storage
<ol type="i">
The localStorage Object
The sessionStorage Object

Web Workers
<ol type="i">
Web Worker File
Web Worker Object
Web Workers and the DOM

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –

use data URIs
use window events
use form events
use keyboard events
use mouse events
use clipboard events
use drag and drop
use geolocation
use web storage
use web workers
</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>28/10/2024</td><td>20/11/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>90</td><td>Module04</td><td>STC CS MSSL-411</td><td>
Node.js
<ol type="1">
Node Basics
<ol type="a">
Managing Node Versions with Node Version Manager
Responding to a Simple Browser Request
Interactively Trying Out Node Code Snippets with REPL
Reading and Writing File Data
Getting Input from the Terminal
Getting the Path to the Current Script
Working with Node Timers and Understanding the Node Event Loop

Node Modules
<ol type="a">
Searching for a Specific Node Module via npm
Converting Your Library into a Node Module
Taking Your Code Across Module Environments
Creating an Installable Node Module
Writing Multiplatform Libraries
Unit Testing Your Modules

Managing Node
<ol type="a">
Using Environment Variables
Managing Callback Hell
Accessing Command-Line Functionality Within a Node Application
Passing Command-Line Arguments
Creating a Command-Line Utility with Help from Commander
Keeping a Node Instance Up and Running
Monitoring Application Changes and Restarting During Local Development
Scheduling Repeat Tasks
Testing the Performance and Capability of Your WebSockets Application

Remote Data
<ol type="a">
Fetching Remote Data
Screen Scraping
Accessing JSON-Formatted Data via a RESTful API

Building Web Applications with Express
<ol type="a">
Using Express to Respond to Requests
Using the Express-Generator
Routing
Working with OAuth
OAuth 2 User Authentication with Passport.js
Serving Up Formatted Data
Building a RESTful API
Building a GraphQL API

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Create web application using Node.js & Express.js</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>29/10/2024</td><td>21/11/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>91</td><td>Module04</td><td>STC CS MSSL-411</td><td>
Node.js
<ol type="1">
Node Basics
<ol type="a">
Managing Node Versions with Node Version Manager
Responding to a Simple Browser Request
Interactively Trying Out Node Code Snippets with REPL
Reading and Writing File Data
Getting Input from the Terminal
Getting the Path to the Current Script
Working with Node Timers and Understanding the Node Event Loop

Node Modules
<ol type="a">
Searching for a Specific Node Module via npm
Converting Your Library into a Node Module
Taking Your Code Across Module Environments
Creating an Installable Node Module
Writing Multiplatform Libraries
Unit Testing Your Modules

Managing Node
<ol type="a">
Using Environment Variables
Managing Callback Hell
Accessing Command-Line Functionality Within a Node Application
Passing Command-Line Arguments
Creating a Command-Line Utility with Help from Commander
Keeping a Node Instance Up and Running
Monitoring Application Changes and Restarting During Local Development
Scheduling Repeat Tasks
Testing the Performance and Capability of Your WebSockets Application

Remote Data
<ol type="a">
Fetching Remote Data
Screen Scraping
Accessing JSON-Formatted Data via a RESTful API

Building Web Applications with Express
<ol type="a">
Using Express to Respond to Requests
Using the Express-Generator
Routing
Working with OAuth
OAuth 2 User Authentication with Passport.js
Serving Up Formatted Data
Building a RESTful API
Building a GraphQL API

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Create web application using Node.js & Express.js</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>30/10/2024</td><td>23/11/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>92</td><td>Module04</td><td>STC CS MSSL-411</td><td>
Node.js
<ol type="1">
Node Basics
<ol type="a">
Managing Node Versions with Node Version Manager
Responding to a Simple Browser Request
Interactively Trying Out Node Code Snippets with REPL
Reading and Writing File Data
Getting Input from the Terminal
Getting the Path to the Current Script
Working with Node Timers and Understanding the Node Event Loop

Node Modules
<ol type="a">
Searching for a Specific Node Module via npm
Converting Your Library into a Node Module
Taking Your Code Across Module Environments
Creating an Installable Node Module
Writing Multiplatform Libraries
Unit Testing Your Modules

Managing Node
<ol type="a">
Using Environment Variables
Managing Callback Hell
Accessing Command-Line Functionality Within a Node Application
Passing Command-Line Arguments
Creating a Command-Line Utility with Help from Commander
Keeping a Node Instance Up and Running
Monitoring Application Changes and Restarting During Local Development
Scheduling Repeat Tasks
Testing the Performance and Capability of Your WebSockets Application

Remote Data
<ol type="a">
Fetching Remote Data
Screen Scraping
Accessing JSON-Formatted Data via a RESTful API

Building Web Applications with Express
<ol type="a">
Using Express to Respond to Requests
Using the Express-Generator
Routing
Working with OAuth
OAuth 2 User Authentication with Passport.js
Serving Up Formatted Data
Building a RESTful API
Building a GraphQL API

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Create web application using Node.js & Express.js</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>31/10/2024</td><td>24/11/2024</td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
93	Module04	STC CS MSSL-412	**jQuery** 1. jQuery: Introduction a. INTRODUCTION b. DIFFERENCE BETWEEN FRAMEWORK AND LIBRARY c. FRAMEWORK i. Benefits d. WHAT IS A LIBRARY? e. WHY DO WE NEED A LIBRARY? f. HISTORY g. NET FRAMEWORK AND LIBRARIES h. SYNTAX TO USE jQuery i. 5WHY DO WE USE jQuery IN THE APP? j. REASONS TO USE jQuery IN YOUR APPLICATION k. DIFFERENCE BETWEEN JavaScript AND jQuery l. 6HOW TO INSTALL jQuery? m. USING jQuery WITH CDN n. DOWNLOAD THE jQuery LIBRARY FROM jQuery.com o. DOWNLOAD THE jQuery LIBRARY USING npm (NODE PACKAGE MANAGER) p. DOWNLOAD THE jQuery LIBRARY USING BOWER q. WHY DO WE USE CDN? i. $(document).ready() function in jQuery ii. ready () function in jQuery r. TYPES OF jQuery METHODS s. jQuery | SELECTION AND EVENT METHODS i. Selector t. jQuery SELECTORS u. jQuery METHODS v. DOM MANIPULATION METHODS w. jQuery EVENTS x. MOUSE EVENTS y. KEYBOARD EVENTS z. FORM EVENTS aa. DOCUMENT EVENTS ab. EVENT MANIPULATION METHODS ac. TRIGGER METHODS ad. BINDING METHODS ae. EVENT ATTRIBUTES i. this Keyword in Event Handler ii. Removing Event Handlers af. jQuery HTML/CSS METHODS ag. MANIPULATE DOM ELEMENT'S DIMENSIONS USING jQuery ah. TRAVERSING DOM ELEMENTS USING jQuery ai. THE DOM aj. CSS MANIPULATION USING jQuery ak. jQuery EFFECTS al. jQuery EFFECT METHODS am. jQuery - ATTRIBUTES MANIPULATION an. STANDARD ATTRIBUTES ao. jQuery - GET DATA ATTRIBUTES ap. jQuery - GET STANDARD ATTRIBUTES aq. jQuery - SET STANDARD ATTRIBUTES ar. jQuery - SET CUSTOM ATTRIBUTES as. jQuery - AJAX at. BENEFITS OF jQuery au. DISADVANTAGES 2. jQuery Project One: To-Do List a. INTRODUCTION b. HTML c. KEY HTML FEATURES i. Head Section ii. Body Section d. CSS e. KEY CSS COMPONENTS f. WHY DO WE USE CSS? g. jQuery PROJECT: TO-DO LIST h. PROJECT EXPLANATION i. jQuery j. WHY DO WE CHOOSE TO USE jQuery? 3. jQuery Project Two: Portfolio a. INTRODUCTION b. HTML c. KEY HTML FEATURES	02/11/2024	25/11/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>
i. Head Section
ii. Body Section
d. CSS
e. KEY CSS COMPONENTS
f. WHY DO WE USE CSS?
g. jQuery PROJECT: PORTFOLIO
h. PROJECT EXPLANATION
4. jQuery Project Third: Flip Card
a. INTRODUCTION
b. HTML
c. KEY HTML FEATURES
i. Head Section
ii. Body Section
d. CSS
e. KEY CSS COMPONENTS
f. WHY DO WE USE CSS?
g. jQuery FILE CODE EXPLANATION
5. Code Optimization
a. INTRODUCTION
b. IMPORTANT TYPES OF CODE ENHANCEMENT
c. CODE OPTIMIZATION AND ORGANIZATION IN jQuery
d. THE jQuery SECURITY MODEL
i. The $() Function
ii. jQuery “XSS Vulnerability”
iii. AJAX for jQuery $ .get ()
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Learn optimized use of jQuery</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td></td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
94	Module04	STC CS MSSL-412	**jQuery** 1. jQuery: Introduction a. INTRODUCTION b. DIFFERENCE BETWEEN FRAMEWORK AND LIBRARY c. FRAMEWORK i. Benefits d. WHAT IS A LIBRARY? e. WHY DO WE NEED A LIBRARY? f. HISTORY g. NET FRAMEWORK AND LIBRARIES h. SYNTAX TO USE jQuery i. 5WHY DO WE USE jQuery IN THE APP? j. REASONS TO USE jQuery IN YOUR APPLICATION k. DIFFERENCE BETWEEN JavaScript AND jQuery l. 6HOW TO INSTALL jQuery? m. USING jQuery WITH CDN n. DOWNLOAD THE jQuery LIBRARY FROM jQuery.com o. DOWNLOAD THE jQuery LIBRARY USING npm (NODE PACKAGE MANAGER) p. DOWNLOAD THE jQuery LIBRARY USING BOWER q. WHY DO WE USE CDN? i. $(document).ready() function in jQuery ii. ready () function in jQuery r. TYPES OF jQuery METHODS s. jQuery | SELECTION AND EVENT METHODS i. Selector t. jQuery SELECTORS u. jQuery METHODS v. DOM MANIPULATION METHODS w. jQuery EVENTS x. MOUSE EVENTS y. KEYBOARD EVENTS z. FORM EVENTS aa. DOCUMENT EVENTS ab. EVENT MANIPULATION METHODS ac. TRIGGER METHODS ad. BINDING METHODS ae. EVENT ATTRIBUTES i. this Keyword in Event Handler ii. Removing Event Handlers af. jQuery HTML/CSS METHODS ag. MANIPULATE DOM ELEMENT'S DIMENSIONS USING jQuery ah. TRAVERSING DOM ELEMENTS USING jQuery ai. THE DOM aj. CSS MANIPULATION USING jQuery ak. jQuery EFFECTS al. jQuery EFFECT METHODS am. jQuery - ATTRIBUTES MANIPULATION an. STANDARD ATTRIBUTES ao. jQuery - GET DATA ATTRIBUTES ap. jQuery - GET STANDARD ATTRIBUTES aq. jQuery - SET STANDARD ATTRIBUTES ar. jQuery - SET CUSTOM ATTRIBUTES as. jQuery - AJAX at. BENEFITS OF jQuery au. DISADVANTAGES 2. jQuery Project One: To-Do List a. INTRODUCTION b. HTML c. KEY HTML FEATURES i. Head Section ii. Body Section d. CSS e. KEY CSS COMPONENTS f. WHY DO WE USE CSS? g. jQuery PROJECT: TO-DO LIST h. PROJECT EXPLANATION i. jQuery j. WHY DO WE CHOOSE TO USE jQuery? 3. jQuery Project Two: Portfolio a. INTRODUCTION b. HTML c. KEY HTML FEATURES	03/11/2024	26/11/2024

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>
i. Head Section
ii. Body Section
d. CSS
e. KEY CSS COMPONENTS
f. WHY DO WE USE CSS?
g. jQuery PROJECT: PORTFOLIO
h. PROJECT EXPLANATION
4. jQuery Project Third: Flip Card
a. INTRODUCTION
b. HTML
c. KEY HTML FEATURES
i. Head Section
ii. Body Section
d. CSS
e. KEY CSS COMPONENTS
f. WHY DO WE USE CSS?
g. jQuery FILE CODE EXPLANATION
5. Code Optimization
a. INTRODUCTION
b. IMPORTANT TYPES OF CODE ENHANCEMENT
c. CODE OPTIMIZATION AND ORGANIZATION IN jQuery
d. THE jQuery SECURITY MODEL
i. The $() Function
ii. jQuery “XSS Vulnerability”
iii. AJAX for jQuery $ .get ()
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Learn optimized use of jQuery</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td></td><td></td></tr>
<tr><td>95</td><td>Module05</td><td>STC CS XADR-501</td><td>
The XML
1. Overview of XML
a. Understanding Markup Languages
b. Entering into XML
c. Goals of XML
d. Recommendations and Standards for XML
2. Basics of XML
a. Structure and Syntax for XML
b. Well-Formed XML
c. Document Classes & DTD
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• understand XML
• create Well-Formed XML using structure and systax of XML
• create DTD</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>04/11/2024</td><td>27/11/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>96</td><td>Module05</td><td>STC CS XADR-502</td><td>ADO.NET Disconnected Classes
1. Working with the DataTable and DataSet Classes
a. The DataTable Class
b. Using DataView as a Window into a Data Table
c. Using a DataSet Object to Coordinate Work Between Data Tables
2. Serialization, Specialized Types, and Data Binding
a. Serializing and Deserializing the Data Table with XML Data
b. Serializing and Deserializing DataSet Objects
c. Handling Specialized Types
d. Data Binding Overview
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• work with DataTable, DataSet classes
• Serialize and Deserialize DataTable and DataSet</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>05/11/2024</td><td>28/11/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Instructor Sick</td><td>05/11/2024</td><td></td></tr>
<tr><td>97</td><td>Module05</td><td>STC CS XADR-502</td><td>ADO.NET Disconnected Classes
1. Working with the DataTable and DataSet Classes
a. The DataTable Class
b. Using DataView as a Window into a Data Table
c. Using a DataSet Object to Coordinate Work Between Data Tables
2. Serialization, Specialized Types, and Data Binding
a. Serializing and Deserializing the Data Table with XML Data
b. Serializing and Deserializing DataSet Objects
c. Handling Specialized Types
d. Data Binding Overview
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• work with DataTable, DataSet classes
• Serialize and Deserialize DataTable and DataSet</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>06/11/2024</td><td>30/11/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>98</td><td>Module05</td><td>STC CS XADR-503</td><td>
ADO.NET Connected Classes
1. Connecting to the Data Store
a. Using Providers to Move Data
b. Getting Started with the DbConnection Object
c. Storing the Connection String in the Application Configuration File
d. Encrypted Communications to SQL Server
e. Storing Encrypted Connection Strings in Web Applications
f. Connection Pooling
2. Reading and Writing Data
a. DbCommand Object
b. DbDataReader Object
c. Using Multiple Active Result Sets (MARS) to Execute Multiple Commands on a Connection
d. Performing Bulk Copy Operations with a SqlBulkCopy Object
e. DbDataAdapter Object
f. DbProviderFactory Classes
g. Using DbException to Catch Provider Exceptions
h. Working with SQL Server User-Defined Types (UDTs)
3. Working with Transactions
a. What Is a Transaction?
b. Concurrency Models and Database Locking
c. Transaction Isolation Levels
d. Single Transactions and Distributed Transactions
e. Creating a Transaction
f. Introducing the System.Transactions Namespace
g. Working with Distributed Transactions
h. Viewing Distributed Transactions
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• work with DbConnection object
• store connection string in the Application Configuration File
• Encrypt connection and store encrypted connection string in web application
• work with DbCommand, DbDataReader, SqlBulkCopy, DbDataAdapter, DbProviderFactory and DbException
• use MARS to execute multiple commands on a connection
• work with SQL Server User-Defined Types
• understand transaction, concurrency models and database locking, transaction isolation levels, single transaction and distributed transactions
• create a transaction
• work with distributed transactions
• view distributed transactions</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>07/11/2024</td><td>01/12/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Instructor Sick</td><td>07/11/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>99</td><td>Module05</td><td>STC CS XADR-503</td><td>ADO.NET Connected Classes
1. Connecting to the Data Store
a. Using Providers to Move Data
b. Getting Started with the DbConnection Object
c. Storing the Connection String in the Application Configuration File
d. Encrypted Communications to SQL Server
e. Storing Encrypted Connection Strings in Web Applications
f. Connection Pooling
2. Reading and Writing Data
a. DbCommand Object
b. DbDataReader Object
c. Using Multiple Active Result Sets (MARS) to Execute Multiple Commands on a Connection
d. Performing Bulk Copy Operations with a SqlBulkCopy Object
e. DbDataAdapter Object
f. DbProviderFactory Classes
g. Using DbException to Catch Provider Exceptions
h. Working with SQL Server User-Defined Types (UDTs)
3. Working with Transactions
a. What Is a Transaction?
b. Concurrency Models and Database Locking
c. Transaction Isolation Levels
d. Single Transactions and Distributed Transactions
e. Creating a Transaction
f. Introducing the System.Transactions Namespace
g. Working with Distributed Transactions
h. Viewing Distributed Transactions
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• work with DbConnection object
• store connection string in the Application Configuration File
• Encrypt connection and store encrypted connection string in web application
• work with DbCommand, DbDataReader, SqlBulkCopy, DbDataAdapter, DbProviderFactory and DbException
• use MARS to execute multiple commands on a connection
• work with SQL Server User-Defined Types
• understand transaction, concurrency models and database locking, transaction isolation levels, single transaction and distributed transactions
• create a transaction
• work with distributed transactions
• view distributed transactions</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>09/11/2024</td><td>02/12/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>sick</td><td>09/11/2024</td><td></td></tr>
<tr><td>0</td><td></td><td></td><td>Exam Review Class</td><td>10/11/2024</td><td></td></tr>
<tr><td>0</td><td></td><td></td><td>External Exam</td><td>11/11/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>100</td><td>Module05</td><td>STC CS XADR-504</td><td>
Crystal Reports
1. Introduction to SAP Crystal Reports 2016
a. About Crystal Reports
b. About the Crystal Reports documentation
i. Locate information quickly
ii. Sample Reports
2. Quick Start
a. Learning how to use Crystal Reports
i. Sample data - Xtreme.mdb
b. Report Creation Wizards
i. Standard
ii. Cross-Tab
iii. Mailing Label
iv. OLAP
c. Quick start for new users
i. Before you begin
ii. Creating the report
iii. Record Selection
iv. Grouping and sorting
v. Completing the report
d. Quick start for advanced users
i. Choosing a report type and data source
ii. Working with report elements on the Design tab
iii. Using other reporting features
3. Report Design Concepts
a. Basic report design
b. Deciding on the content of the report
i. Stating the purpose
ii. Determining the layout of the report
iii. Finding the data
iv. Manipulating the data
v. Determining printing area characteristics
c. Developing a prototype on paper
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create crystal reports
• understand report design concepts</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>12/11/2024</td><td>03/12/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Unavailable</td><td>12/11/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>101</td><td>Module05</td><td>STC CS XADR-504</td><td>
Crystal Reports
4. Introduction to Reporting
a. Report creation options
b. Choosing data sources and database fields
i. The Database Expert
ii. The Field Explorer
c. About the report design environment
i. Design tab
ii. Preview tab
iii. HTML Preview tab
d. Creating a new report
i. Selecting the data source
ii. Adding tables
iii. Linking multiple tables
iv. Placing data on the report
v. BLOB (Binary Large Object) fields
vi. Formatting data
vii. Record selection
viii. Grouping, sorting, and summarizing data
ix. Using the drill-down option on summarized data
x. Using the zoom feature
xi. Inserting page headers and footers
xii. Adding a title page to the report
xiii. Adding summary information to the report
e. Exploring reports and working with multiple reports
i. The Report Explorer
ii. The Workbench
iii. The Dependency Checker
iv. Opening and docking explorers
f. Beyond basic reports
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create crystal reports
• understand report design environment
• work with multiple reports</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>13/11/2024</td><td>04/12/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Unavailable</td><td>13/11/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>102</td><td>Module06</td><td>STC CS MVC 5-601</td><td>ASP.NET MVC controllers
1. Routing incoming requests
a. Simulating the ASP.NET MVC runtime
b. The URL routing HTTP module
c. Application routes
2. The controller class
a. Aspects of a controller
b. Writing controller classes
c. Processing input data
d. Producing action results

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding the ASP.NET MVC runtime, url routing http module, application routes
• Understanding controller, writing controller, processing input data and producing action results
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>14/11/2024</td><td>05/12/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Unavailable</td><td>14/11/2024</td><td></td></tr>
<tr><td>103</td><td>Module06</td><td>STC CS MVC 5-601</td><td>ASP.NET MVC controllers
1. Routing incoming requests
a. Simulating the ASP.NET MVC runtime
b. The URL routing HTTP module
c. Application routes
2. The controller class
a. Aspects of a controller
b. Writing controller classes
c. Processing input data
d. Producing action results

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding the ASP.NET MVC runtime, url routing http module, application routes
• Understanding controller, writing controller, processing input data and producing action results
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>16/11/2024</td><td>07/12/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Unavailable</td><td>16/11/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>104</td><td>Module06</td><td>STC CS MVC 5-602</td><td>ASP.NET MVC views
1. The structure and behavior of a view engine
a. The mechanics of a view engine
b. Definition of the view template
2. HTML helpers
a. Basic helpers
b. Templated helpers
c. Custom helpers
3. The Razor view engine
a. Inside the view engine
b. Designing a sample view
4. Coding the view
a. Modeling the view
b. Advanced features
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding the structure and behavior of view engine and view templates
• How to write & use HTML helpers
• Understand the razor view engine
• Coding the view
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>17/11/2024</td><td>08/12/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Unavailable</td><td>17/11/2024</td><td></td></tr>
<tr><td>105</td><td>Module06</td><td>STC CS MVC 5-602</td><td>ASP.NET MVC views
1. The structure and behavior of a view engine
a. The mechanics of a view engine
b. Definition of the view template
2. HTML helpers
a. Basic helpers
b. Templated helpers
c. Custom helpers
3. The Razor view engine
a. Inside the view engine
b. Designing a sample view
4. Coding the view
a. Modeling the view
b. Advanced features
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding the structure and behavior of view engine and view templates
• How to write & use HTML helpers
• Understand the razor view engine
• Coding the view
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>18/11/2024</td><td>09/12/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Unavailable</td><td>18/11/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>106</td><td>Module06</td><td>STC CS MVC 5-603</td><td>The model-binding architecture
1. The input model
a. Evolving from the Web Forms input processing
b. Input processing in ASP.NET MVC
2. Model binding
a. Model-binding infrastructure
b. The default model binder
c. Customizable aspects of the default binder
3. Advanced model binding
a. Custom type binders
b. A sample DateTime model binder
<h1>EVIDENCE GUIDANCE</h1><table><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Input processing in ASP.NET MVC
• Bind model and customize default model binder</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>19/11/2024</td><td>12/12/2024</td></tr>
<tr><td>0</td><td></td><td></td><td>Unavailable</td><td>19/11/2024</td><td></td></tr>
<tr><td>107</td><td>Module06</td><td>STC CS MVC 5-603</td><td>The model-binding architecture
1. The input model
a. Evolving from the Web Forms input processing
b. Input processing in ASP.NET MVC
2. Model binding
a. Model-binding infrastructure
b. The default model binder
c. Customizable aspects of the default binder
3. Advanced model binding
a. Custom type binders
b. A sample DateTime model binder
<h1>EVIDENCE GUIDANCE</h1><table><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Input processing in ASP.NET MVC
• Bind model and customize default model binder</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>20/11/2024</td><td>14/12/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>108</td><td>Module06</td><td>STC CS MVC 5-604</td><td>Input forms
1. General patterns of data entry
a. A classic Select-Edit-Post scenario
b. Applying the Post-Redirect-Get pattern
2. Automating the writing of input forms
a. Predefined display and editor templates
b. Custom templates for model data types
3. Input validation
a. Using data annotations
b. Advanced data annotations
c. Self-validation

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding and using predefined display and editor templates, custom template for model data type
• Understanding and using data annotations, creating advanced data annotations, implementing self-validation
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>21/11/2024</td><td>15/12/2024</td></tr>
<tr><td>109</td><td>Module06</td><td>STC CS MVC 5-604</td><td>Input forms
1. General patterns of data entry
a. A classic Select-Edit-Post scenario
b. Applying the Post-Redirect-Get pattern
2. Automating the writing of input forms
a. Predefined display and editor templates
b. Custom templates for model data types
3. Input validation
a. Using data annotations
b. Advanced data annotations
c. Self-validation

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding and using predefined display and editor templates, custom template for model data type
• Understanding and using data annotations, creating advanced data annotations, implementing self-validation
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>23/11/2024</td><td>17/12/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>110</td><td>Module06</td><td>STC CS MVC 5-605</td><td>Aspects of ASP.NET MVC applications
1. ASP.NET intrinsic objects
a. HTTP response and SEO
b. Managing the session state
c. Caching data
2. Error handling
a. Handling program exceptions
b. Global error handling
c. Dealing with missing content
3. Localization
a. Using localizable resources
b. Dealing with localizable applications
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding http response and SEO
• Managing session state and caching data
• Handling errors and missing content
• Using localizable resources
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>24/11/2024</td><td>18/12/2024</td></tr>
<tr><td>111</td><td>Module06</td><td>STC CS MVC 5-606</td><td>Securing your application
1. Security in ASP.NET MVC
a. Authentication and authorization
b. Separating authentication from authorization
2. Implementing a membership system
a. Defining a membership controller
b. The Remember-Me feature and Ajax
3. External authentication services
a. The OpenID protocol
b. Authenticating via social networks
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding and implementing authentication and authorization, separating authentication from authorization
• Understanding and implementing membership
• Understanding OpenID protocol and authenticating via social networks
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>25/11/2024</td><td>19/12/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>112</td><td>Module06</td><td>STC CS MVC 5-607</td><td>Design considerations for ASP.NET MVC controllers
1. Shaping up your controller
a. Choosing the right stereotype
b. Fat-free controllers
2. Connecting the presentation and back end
a. The Layered Architecture pattern
b. Injecting data and services in layers
c. Gaining control of the controller factory
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Choosing right stereotype for a controller
• Understanding and using layered architecture pattern, injecting data, services in the layers
• Controlling the controller factory
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>26/11/2024</td><td>21/12/2024</td></tr>
<tr><td>113</td><td>Module06</td><td>STC CS MVC 5-607</td><td>Design considerations for ASP.NET MVC controllers
1. Shaping up your controller
a. Choosing the right stereotype
b. Fat-free controllers
2. Connecting the presentation and back end
a. The Layered Architecture pattern
b. Injecting data and services in layers
c. Gaining control of the controller factory
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Choosing right stereotype for a controller
• Understanding and using layered architecture pattern, injecting data, services in the layers
• Controlling the controller factory
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>27/11/2024</td><td>22/12/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>114</td><td>Module06</td><td>STC CS MVC 5-608</td><td>Customizing ASP.NET MVC controllers
1. The extensibility model of ASP.NET MVC
a. The provider-based model
b. The Service Locator pattern
2. Adding aspects to controllers
a. Action filters
b. Gallery of action filters
c. Special filters
d. Building a dynamic loader filter
3. Action result types
a. Built-in action result types
b. Custom result types

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding provider-based model and the service locator pattern
• Understanding, building and implementing all types of filters to the controller
• Understanding built-in action result types and customizing result type
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>28/11/2024</td><td>23/12/2024</td></tr>
<tr><td>115</td><td>Module06</td><td>STC CS MVC 5-608</td><td>Customizing ASP.NET MVC controllers
1. The extensibility model of ASP.NET MVC
a. The provider-based model
b. The Service Locator pattern
2. Adding aspects to controllers
a. Action filters
b. Gallery of action filters
c. Special filters
d. Building a dynamic loader filter
3. Action result types
a. Built-in action result types
b. Custom result types

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understanding provider-based model and the service locator pattern
• Understanding, building and implementing all types of filters to the controller
• Understanding built-in action result types and customizing result type
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>30/11/2024</td><td>24/12/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>116</td>
<td>Module06</td>
<td>STC CS MVC 5-609</td>
<td>
Testing and testability in ASP.NET MVC

Testability and design
<ol type="a">
DfT
Loosen up your design

The basics of unit testing
<ol type="a">
Working with a test harness
Aspects of testing

Testing your ASP.NET MVC code
<ol type="a">
Which part of your code should you test?
Unit testing ASP.NET MVC code
Dealing with dependencies
Mocking the HTTP context

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Understanding and implementing unit test, dealing with dependencies, mocking the HTTP context</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>01/12/2024</td>
<td>26/12/2024</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>117</td><td>Module07</td><td>STC CS EF6-701</td><td>
Getting Started with Entity Framework 6 Code First using MVC 5
1. Creating an Entity Framework Data Model
a. Create an MVC Web Application
b. Set Up the Site Style
c. Install Entity Framework 6
d. Create the Data Model
e. Create the Database Context
f. Set up EF to initialize the database with test data
g. Set up EF to use a SQL Server Express LocalDB database
h. Creating a Student Controller and Views
i. View the Database
j. Conventions
2. Implementing Basic CRUD Functionality with the Entity Framework in ASP.NET MVC Application
a. Create a Details Page
b. Update the Create Page
c. Update the Edit HttpPost Page
d. Updating the Delete Page
e. Ensuring that Database Connections Are Not Left Open
f. Handling Transactions
3. Sorting, Filtering, and Paging with the Entity Framework in an ASP.NET MVC Application
a. Add Column Sort Links to the Students Index Page
b. Add a Search Box to the Students Index Page
c. Add Paging to the Students Index Page
d. Create an About Page That Shows Student Statistics
<table>
<tr><td colspan="2">EVIDENCE GUIDANCE</td></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand Entity Data Model
• Install Entity Framework 6
• Create Data Model
• Create Database Context
• Setup Entity Framework to initialize database
• Setup Entity Framework to use LocalDB
• Implementing CRUD operation using Entity Framework
• Ensure database connection is not open
• Implement Transaction using Entity Framework
• Filter, Sort, Pagination with Entity Framework</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>02/12/2024</td><td>28/12/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>118</td><td>Module07</td><td>STC CS EF6-701</td><td>
Getting Started with Entity Framework 6 Code First using MVC 5
4. Connection Resiliency and Command Interception with the Entity Framework in an ASP.NET MVC Application
a. Enable connection resiliency
b. Enable Command Interception
c. Test logging and connection resiliency
5. Code First Migrations and Deployment with the Entity Framework in an ASP.NET MVC Application
a. Enable Code First Migrations
b. Deploy to Windows Azure
c. Advanced Migrations Scenarios
d. Code First Initializers
6. Creating a More Complex Data Model for an ASP.NET MVC Application
a. Customize the Data Model by Using Attributes
b. Complete Changes to the Student Entity
c. Create the Instructor Entity
d. Create the OfficeAssignment Entity
e. Modify the Course Entity
f. Create the Department Entity
g. Modify the Enrollment Entity
h. Entity Diagram Showing Relationships
i. Customize the Data Model by adding Code to the Database Context
j. Seed the Database with Test Data
k. Add a Migration and Update the Database
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Enable connection resiliency
• Enable command interception
• Enable Code First Migrations
• Code First initializes
• Customize the Data Model by Using Attributes
• Entity Diagram Showing Relationships
• Customize the Data Model by adding Code to the Database Context
• Seed the Database with Test Data
• Add a Migration and Update the Database</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>03/12/2024</td><td>29/12/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>119</td><td>Module07</td><td>STC CS EF6-701</td><td>Getting Started with Entity Framework 6 Code First using MVC 5
7. Reading Related Data with the Entity Framework in an ASP.NET MVC Application
a. Lazy, Eager, and Explicit Loading of Related Data
b. Create a Courses Page That Displays Department Name
c. Create an Instructors Page That Shows Courses and Enrollments
8. Updating Related Data with the Entity Framework in an ASP.NET MVC Application
a. Customize the Create and Edit Pages for Courses
b. Adding an Edit Page for Instructors
c. Adding Course Assignments to the Instructor Edit Page
d. Update the DeleteConfirmed Method
e. Add office location and courses to the Create page
f. Handling Transactions
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Lazy, Eagerand Explicit Loading of Related Data
• Update Related Data
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>04/12/2024</td><td>30/12/2024</td></tr>
<tr><td>120</td><td>Module07</td><td>STC CS EF6-701</td><td>Getting Started with Entity Framework 6 Code First using MVC 5
9. Async and Stored Procedures with the Entity Framework in an ASP.NET MVC Application
a. Why bother with asynchronous code
b. Create the Department controller
c. Use stored procedures for inserting, updating, and deleting
d. Deploy to Windows Azure
10. Handling Concurrency with the Entity Framework 6 in an ASP.NET MVC 5 Application
a. Concurrency Conflicts
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Write asynchronous code with Entity Framework
• Use stored procedure with Entity Framework
• Handling Concurrency with Entity Framework
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>05/12/2024</td><td>31/12/2024</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>121</td><td>Module07</td><td>STC CS EF6-701</td><td>
Getting Started with Entity Framework 6 Code First using MVC 5
11. Implementing Inheritance with the Entity Framework 6 in an ASP.NET MVC 5 Application
a. Options for mapping inheritance to database tables
b. Create the Person class
c. Make Student and Instructor classes inherit from Person
d. Add the Person Entity Type to the Model
e. Create and Update a Migrations File
f. Testing
g. Deploy to Windows Azure
12. Advanced Entity Framework 6 Scenarios for an MVC 5 Web Application
a. Performing Raw SQL Queries
b. No-Tracking Queries
c. Examining SQL sent to the database
d. Repository and unit of work patterns
e. Proxy classes
f. Automatic change detection
g. Automatic validation
h. Entity Framework Power Tools
i. Entity Framework source code
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Implement inheritance with Entity Framework
• Perform Raw SQL Query
• No-Tracking Query
• Repository and unit of work pattern
• Proxy class
• Automatic change detection
• Automatic validation
• User Entity Framework power tools
• Entity Framework source code</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>07/12/2024</td><td>01/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>122</td>
<td>Module08</td>
<td>STC CS MVC 5 Web API-801</td>
<td>
First Web API Application and Essential Techniques
1. Your First Web API Application
a. Preparing the Example Project
i. Creating the Visual Studio Project
ii. Adding and Updating NuGet Packages
iii. Setting the Port and Start URL
b. Creating the MVC Application
i. Creating the Model
ii. Creating the MVC Controller
iii. Creating the Views
iv. Using the MVC Application
c. Creating the Web Service
i. Creating the Web API Controller
ii. Testing the Web API Controller
d. Implementing the Single-Page Client
i. Setting Up JavaScript IntelliSense
ii. Defining the Client-Side Data Model and Controller
iii. Registering the JavaScript File
iv. Applying Data Bindings
v. Testing the Single-Page Client
e. Measuring the Single-Page Implementation
2. Essential Techniques
a. Preparing the Example Project
i. Adding and Updating NuGet Packages
ii. Creating the Web API Controller
iii. Creating the MVC Framework Controller
iv. Setting the Port and Start URL
v. Testing the Web Service
b. Understanding Asynchronous Methods
i. Understanding the Problem Asynchronous Methods Solve
ii. Implementing an Asynchronous Interface
iii. Creating a Self-Contained Asynchronous Method Body
iv. Returning a Task from a Synchronous Method Body
c. Making Ajax Requests with jQuery
i. Making an Ajax Request
ii. Understanding the $.ajax Method
d. Using Knockout
i. Applying the Bindings
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand the ASP.NET MVC 5 web API & it's environment
• Creating basic ASP.NET MVC 5 web API
• Understanding essential techniques</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>08/12/2024</td>
<td>02/01/2025</td>
</tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
123	Module08	STC CS MVC 5 Web API-802	**Understanding HTTP Web Services, Preparing and Deploying the Example Project** 1. Understanding HTTP Web Services a. Understanding ASP.NET Web API i. Understanding Single-Page Applications ii. Understanding Native Applications iii. Understanding Shared-Model Applications iv. Understanding Service Applications b. Understanding Simple Web Services c. Understanding RESTful Web Services i. Embracing HTTP ii. Adding Data Discovery 2. SportsStore: Preparation a. Preparing the Example Project i. Adding and Updating NuGet Packages ii. Creating a Prep Controller iii. Creating a Razor Layout iv. Creating the OWIN Startup Class v. Setting the TCP Port b. Creating the Product and Order Models i. Defining the Model Classes ii. Creating the Repository Classes iii. Testing the Repository iv. Checking the Database Schema c. Configuring ASP.NET Identity i. Defining the User and Role Classes ii. Creating the Database Context Classes iii. Creating the Manager Classes iv. Adding the Configuration Statements v. Testing ASP.NET Identity vi. Removing the Application Cookie 3. SportsStore: A RESTful Application a. Creating a RESTful Web Service i. Testing the Products Web Service b. Putting the Web Service in Context i. Working with Regular C# Objects ii. Using the RESTful Action Method Convention c. Configuring Serialization d. Adding Basic Data Validation e. Securing the Product Web Service i. Restricting Access to Action Method ii. Authenticating Requests f. Adding Model Validation i. Applying Validation Attributes ii. Validating the Model g. Adding Simple Dependency Injection i. Recapping the Problem ii. Creating the Dependency Resolver iii. Using the Dependency Resolver in the Controller Class h. Creating a Non-RESTful Web Service i. Preparing the Routing Configuration ii. Preparing the Model Objects iii. Preventing Formatting Loops iv. Defining the Web API Controller i. Completing the Product Controller 4. SportsStore: Creating the Clients a. Preparing the Example Project i. Setting Up JavaScript IntelliSense ii. Updating the Layout b. Implementing the Common Code i. Defining the Ajax Layer ii. Defining the Model iii. Defining the Authentication Controller iv. Defining the Products Controller v. Defining the Orders Controller c. Creating the Customer Client i. Creating the Customer Model ii. Creating the Customer Controller iii. Creating the Views d. Creating the Administration Client i. Creating the Admin Model	09/12/2024	04/01/2025

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>ii. Creating the Admin Controller
iii. Creating the Views
iv. Testing the Admin Client
5. SportsStore: Deployment
a. Preparing the SportsStore Application
i. Preventing the Product Database from Resetting
ii. Adding Database Connection Strings
b. Preparing Azure
i. Creating the Databases
ii. Creating the Web Site
iii. Downloading the Publish Profile
c. Deploying the Application
i. Configuring the Databases
d. Publishing the Application
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand HTTP web services
• Preparing and Deploying the Example Project</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td></td><td></td></tr>
<tr><td>0</td><td></td><td></td><td>Exam Review Class</td><td>10/12/2024</td><td></td></tr>
<tr><td>0</td><td></td><td></td><td>Exam</td><td>11/12/2024</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>124</td><td>Module08</td><td>STC CS MVC 5 Web API-803</td><td>
Anatomy, Create and Configure Web API
1. The Anatomy of ASP.NET Web API
a. Understanding the Web API Namespaces and Types
b. Understanding the Web API Context Objects
i. Getting Information About the Request
ii. Getting Information About the Controller
iii. Getting Information About Everything Else
c. Understanding the Web API Components
i. Application Configuration
ii. Controllers, Actions, and Results
iii. Services
iv. Dispatchers and Handlers
2. Creating and Configuring a Web API Application
a. Preparing the Example Project
i. Creating the Model and Repository
ii. Creating an HTTP Web Service
iii. Creating the Browser Client
iv. Testing the Example Application
b. Configuring a Web API Application
i. Configuring Web API Through the ASP.NET Platform
ii. Understanding the Configuration Object
c. Configuring Web API Dependency Injection
i. Preparing for Dependency Injection
ii. Understanding the Web API Dependency Interfaces
iii. Installing the Dependency Injection Container
iv. Implementing the Dependency Interfaces
v. Configuring Web API
d. Configuring Dependency Injection for Web API and MVC
i. Declaring the Dependency
ii. Installing the Dependency Injection Packages
iii. Adding MVC Support to the Resolver
iv. Configuring the MVC Framework
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand Web API Namespaces, Types, Context Objects, Components
• Able to create and configure Web API and Dependency Injection for Web API and MVC</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>12/12/2024</td><td>05/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>125</td><td>Module08</td><td>STC CS MVC 5 Web API-803</td><td>
Anatomy, Create and Configure Web API
1. The Anatomy of ASP.NET Web API
a. Understanding the Web API Namespaces and Types
b. Understanding the Web API Context Objects
i. Getting Information About the Request
ii. Getting Information About the Controller
iii. Getting Information About Everything Else
c. Understanding the Web API Components
i. Application Configuration
ii. Controllers, Actions, and Results
iii. Services
iv. Dispatchers and Handlers
2. Creating and Configuring a Web API Application
a. Preparing the Example Project
i. Creating the Model and Repository
ii. Creating an HTTP Web Service
iii. Creating the Browser Client
iv. Testing the Example Application
b. Configuring a Web API Application
i. Configuring Web API Through the ASP.NET Platform
ii. Understanding the Configuration Object
c. Configuring Web API Dependency Injection
i. Preparing for Dependency Injection
ii. Understanding the Web API Dependency Interfaces
iii. Installing the Dependency Injection Container
iv. Implementing the Dependency Interfaces
v. Configuring Web API
d. Configuring Dependency Injection for Web API and MVC
i. Declaring the Dependency
ii. Installing the Dependency Injection Packages
iii. Adding MVC Support to the Resolver
iv. Configuring the MVC Framework
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand Web API Namespaces, Types, Context Objects, Components
• Able to create and configure Web API and Dependency Injection for Web API and MVC</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>14/12/2024</td><td>06/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>126</td><td>Module08</td><td>STC CS MVC 5 Web API-804</td><td>
Action Method Results and Media Type Formatters
1. Action Method Results
a. Preparing the Example Project
b. Understanding Action Method Results
c. Returning No Result
i. Consuming a No Result Action Method
d. Returning an Action Result
i. Understanding the IHttpActionResult Interface
ii. Using the ApiController Action Result Methods
iii. Returning Other Status Codes
iv. Creating a Custom Action Result
e. Returning Model Data
i. Understanding the Default Behavior
ii. Understanding the Content Negotiation Process
iii. Implementing a Custom Content Negotiator
iv. Bypassing Content Negotiation
f. Returning Negotiable Action Results
i. Creating Negotiable Action Results
2. Creating Media Type Formatters
a. Preparing the Example Project
b. Creating a Media Type Formatter
i. Implementing a Basic Media Type Formatter
c. Refining the Custom Formatter
i. Supporting Content Encodings
ii. Setting the HTTP Response Headers
d. Participating in the Negotiation Process
i. Creating a Media Type Mapping
ii. Using the Mapping Extension Methods
e. Creating Per-Request Media Type Formatters
i. Creating the Formatter Instance
ii. Testing the Per-Request Formatter
3. Using the Built-in Media Formatters
a. Preparing the Example Project
b. Working with the Built-in Media Type Formatters
i. Listing the Built-in Media Type Formatters
ii. Dealing with Type Matching During Negotiation
c. Working with the JSON Media Type Formatter
i. Configuring the JSON Media Type Formatter
ii. Configuring Json.Net
d. Using the XML Media Type Formatter
i. Configuring the XML Media Type Formatter
ii. Getting the Xml Media Type Formatter Working
<table>
<tr><td colspan="2">EVIDENCE GUIDENCE</td></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand and able to return action results
• Able to create and use media formatters</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>15/12/2024</td><td>07/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>127</td><td>Module08</td><td>STC CS MVC 5 Web API-804</td><td>
Action Method Results and Media Type Formatters
1. Action Method Results
a. Preparing the Example Project
b. Understanding Action Method Results
c. Returning No Result
i. Consuming a No Result Action Method
d. Returning an Action Result
i. Understanding the IHttpActionResult Interface
ii. Using the ApiController Action Result Methods
iii. Returning Other Status Codes
iv. Creating a Custom Action Result
e. Returning Model Data
i. Understanding the Default Behavior
ii. Understanding the Content Negotiation Process
iii. Implementing a Custom Content Negotiator
iv. Bypassing Content Negotiation
f. Returning Negotiable Action Results
i. Creating Negotiable Action Results
2. Creating Media Type Formatters
a. Preparing the Example Project
b. Creating a Media Type Formatter
i. Implementing a Basic Media Type Formatter
c. Refining the Custom Formatter
i. Supporting Content Encodings
ii. Setting the HTTP Response Headers
d. Participating in the Negotiation Process
i. Creating a Media Type Mapping
ii. Using the Mapping Extension Methods
e. Creating Per-Request Media Type Formatters
i. Creating the Formatter Instance
ii. Testing the Per-Request Formatter
3. Using the Built-in Media Formatters
a. Preparing the Example Project
b. Working with the Built-in Media Type Formatters
i. Listing the Built-in Media Type Formatters
ii. Dealing with Type Matching During Negotiation
c. Working with the JSON Media Type Formatter
i. Configuring the JSON Media Type Formatter
ii. Configuring Json.Net
d. Using the XML Media Type Formatter
i. Configuring the XML Media Type Formatter
ii. Getting the Xml Media Type Formatter Working
<table>
<tr><td colspan="2">EVIDENCE GUIDANCE</td></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand and able to return action results
• Able to create and use media formatters</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>17/12/2024</td><td>08/01/2025</td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
128	Module08	STC CS MVC 5 Web API-805	**Parameter, Model Binding, Data Binding & Model Validation** 1. Understanding Parameter and Model Binding a. Preparing the Example Project i. Creating the Controller ii. Creating the Client iii. Adding a New Route iv. Testing the Example Application b. Understanding the Default Binding Behavior i. Understanding Parameter Binding ii. Understanding Model Binding c. Performing Binding Customizations i. Binding Complex Types from the Request URL ii. Binding Simple Types from the Request Body iii. Defining a Binding Rule d. Manually Obtaining Request Values i. Handling POST Requests 2. Binding Simple Data Types a. Preparing the Example Project b. Preparing the Common Code c. Working with Value Providers and Value Provider Factories i. Understanding Value Providers and Value Provider Factories ii. Creating a Custom Value Provider and Factory d. Applying a Custom Value Provider and Factory i. Understanding How Web API Looks for Values ii. Applying a Value Provider Factory with an Attribute iii. Extending the Default Behavior iv. Creating a Parameter Binding Rule 3. Binding Complex Data Types Part I a. Preparing the Example Project b. Using the Built-in Model Binders i. Binding Objects ii. Broadening the Source of Binding Values iii. Binding Collections and Arrays iv. Binding Key-Value Pairs c. Working with Custom Model Binders i. Preparing the Application ii. Understanding Model Binders iii. Creating a Custom Model Binder iv. Applying a Custom Model Binder d. Using Type Converters i. Understanding Type Converters ii. Creating a Type Converter iii. Applying a Type Converter 4. Binding Complex Data Types Part II a. Preparing the Example Project i. Testing the Application b. Creating a Custom Media Type Formatter i. Preparing the Client ii. Creating the Media Type Formatter iii. Registering and Testing the Media Type Formatter c. Using the Built-in Media Type Formatters i. Handling URL-Encoded Data ii. Handling JSON Requests iii. Handling XML Requests d. Customizing the Model Binding Process i. Changing the Behavior of the Default Action Value Binder ii. Creating a Custom Action Value Binder 5. Model Validation a. Preparing the Example Project i. Testing the Changes b. Understanding Common Data Problems i. Understanding Under-Posting ii. Understanding Over-Posting iii. Understanding Bad Data c. Using Web API Model Validation i. Understanding Model State ii. Testing the Model State iii. Removing the Debug Output Code d. Using the Binding Control Attributes e. Performing Validation with Validation Attributes	18/12/2024	09/01/2025

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>i. Using the Built-in Validation Attributes
f. Creating a Self-validating Model Class
g. Performing Validation in a Media Type Formatter
i. Creating a Validating Media Type Formatter
ii. Registering and Using the Custom Media Type Formatter

<table><tr><td colspan="2">EVIDENCE GUIDANCE</td></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Parameter, Model Binding, Data Binding & Model Validation
Able to create and use parameter, bind model, data and validate model</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td></td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
129	Module08	STC CS MVC 5 Web API-805	**Parameter, Model Binding, Data Binding & Model Validation** 1. Understanding Parameter and Model Binding a. Preparing the Example Project i. Creating the Controller ii. Creating the Client iii. Adding a New Route iv. Testing the Example Application b. Understanding the Default Binding Behavior i. Understanding Parameter Binding ii. Understanding Model Binding c. Performing Binding Customizations i. Binding Complex Types from the Request URL ii. Binding Simple Types from the Request Body iii. Defining a Binding Rule d. Manually Obtaining Request Values i. Handling POST Requests 2. Binding Simple Data Types a. Preparing the Example Project b. Preparing the Common Code c. Working with Value Providers and Value Provider Factories i. Understanding Value Providers and Value Provider Factories ii. Creating a Custom Value Provider and Factory d. Applying a Custom Value Provider and Factory i. Understanding How Web API Looks for Values ii. Applying a Value Provider Factory with an Attribute iii. Extending the Default Behavior iv. Creating a Parameter Binding Rule 3. Binding Complex Data Types Part I a. Preparing the Example Project b. Using the Built-in Model Binders i. Binding Objects ii. Broadening the Source of Binding Values iii. Binding Collections and Arrays iv. Binding Key-Value Pairs c. Working with Custom Model Binders i. Preparing the Application ii. Understanding Model Binders iii. Creating a Custom Model Binder iv. Applying a Custom Model Binder d. Using Type Converters i. Understanding Type Converters ii. Creating a Type Converter iii. Applying a Type Converter 4. Binding Complex Data Types Part II a. Preparing the Example Project i. Testing the Application b. Creating a Custom Media Type Formatter i. Preparing the Client ii. Creating the Media Type Formatter iii. Registering and Testing the Media Type Formatter c. Using the Built-in Media Type Formatters i. Handling URL-Encoded Data ii. Handling JSON Requests iii. Handling XML Requests d. Customizing the Model Binding Process i. Changing the Behavior of the Default Action Value Binder ii. Creating a Custom Action Value Binder 5. Model Validation a. Preparing the Example Project i. Testing the Changes b. Understanding Common Data Problems i. Understanding Under-Posting ii. Understanding Over-Posting iii. Understanding Bad Data c. Using Web API Model Validation i. Understanding Model State ii. Testing the Model State iii. Removing the Debug Output Code d. Using the Binding Control Attributes e. Performing Validation with Validation Attributes	19/12/2024	13/01/2025

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>i. Using the Built-in Validation Attributes
f. Creating a Self-validating Model Class
g. Performing Validation in a Media Type Formatter
i. Creating a Validating Media Type Formatter
ii. Registering and Using the Custom Media Type Formatter

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Parameter, Model Binding, Data Binding & Model Validation
• Able to create and use parameter, bind model, data and validate model
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td></td><td></td></tr>
<tr><td>130</td><td>Module08</td><td>STC CS MVC 5 Web API-806</td><td>Dispatching Requests and URL Routing
1. Dispatching Requests
a. Preparing the Example Project
i. Creating the Model Class
ii. Creating the Web API Web Service
iii. Creating the MVC Controller and View
iv. Testing the Example Application
b. Understanding Request Dispatching
i. Understanding the HttpServer Class
ii. Understanding the HttpRoutingDispatcher Class
iii. Understanding the HttpControllerDispatcher Class
c. Customizing the Dispatch Process
i. Creating Custom Message Handlers
ii. Customizing Other Dispatch Components
2. URL Routing: Part I
a. Preparing the Example Project
i. Testing the Application Changes
b. Understanding URL Routing
i. Understanding the Routing Classes and Interfaces
c. Working with Convention-Based Routing
i. Using Route Templates
ii. Routing to the New Controller
d. Controlling Route Matching
i. Using Routing Data Default Values
ii. Using Routing Constraints
3. URL Routing: Part II
a. Preparing the Example Project
b. Understanding Direct Routing
i. Creating a Direct Route
ii. Creating a Controller-wide Direct Route
c. Customizing URL Routing
i. Using a Route-Specific Message Handler
ii. Applying Custom Constraints to Direct Routes

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understand request dispatching, URL routing, direct routing
• Able to customizing the dispatch process
• Able to work with convention-based routing
• Able to customize URL routing
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>21/12/2024</td><td>14/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>131</td><td>Module08</td><td>STC CS MVC 5 Web API-807</td><td>
API Controllers, Actions & Filters
1. Controllers and Actions
a. Preparing the Example Project
b. Understanding Controllers
i. Creating a Controller
ii. Using Built-in Services and Features
c. Understanding the ApiController Dispatch Process
i. Preparing the Example Controller
ii. Understanding the Action Selection Process
iii. Understanding the RESTful/Non-RESTful Routing Problem
iv. Understanding Filters
v. Understanding the Action Method Execution Process
d. Customizing the Controller Dispatch Process
i. Creating a Custom IHttpActionInvoker Implementation
ii. Creating a Custom IHttpActionSelector Implementation
iii. Creating a Controller-Specific Configuration
2. Filters Part I
a. Preparing the Example Project
b. Understanding Filters
c. Working with Action Filters
i. Creating an Action Filter by Implementing IActionFilter
ii. Using the Convenience Action Filter Base Class
iii. Creating a Short-Circuiting Action Filter
d. Understanding the Filter Pipeline
i. Displaying the Filter Pipeline
ii. Understanding Filter Scope
e. Working with Authentication Filters
i. Preparing for Authentication
ii. Understanding Authentication Filters
iii. Creating an Authentication Filter
iv. Viewing the Filter Pipeline
3. Filters Part II
a. Preparing the Example Project
b. Reviewing Filters in the Dispatch Process
c. Working with Authorization Filters
i. Understanding Authorization Filters
ii. Creating an Authorization Filter
iii. Appling the Authorization Filter
iv. Removing the Authentication Filter
v. Using the Built-in Authorization Filter Attributes
vi. Reworking the Authentication Filter
d. Working with Exception Filters
i. Understanding the Default Behavior
ii. Understanding Exception Filters
iii. Creating an Exception Filter
iv. Deriving the Filter from the ExceptionFilterAttribute Class
e. Working with Override Filters
i. Overriding Built-in Filter Types
ii. Redefining Filter Policies

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understand controllers, ApiController dispatch process, filters, filter pipeline
• Able to customize the controller Dispatch Process
• Able to work with action filters, authentication filters, exception filters, override filters
• Able to customize URL routing
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.
</td><td>22/12/2024</td><td>15/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>132</td>
<td>Module08</td>
<td>STC CS MVC 5 Web API-807</td>
<td>
API Controllers, Actions & Filters
1. Controllers and Actions
a. Preparing the Example Project
b. Understanding Controllers
i. Creating a Controller
ii. Using Built-in Services and Features
c. Understanding the ApiController Dispatch Process
i. Preparing the Example Controller
ii. Understanding the Action Selection Process
iii. Understanding the RESTful/Non-RESTful Routing Problem
iv. Understanding Filters
v. Understanding the Action Method Execution Process
d. Customizing the Controller Dispatch Process
i. Creating a Custom IHttpActionInvoker Implementation
ii. Creating a Custom IHttpActionSelector Implementation
iii. Creating a Controller-Specific Configuration
2. Filters Part I
a. Preparing the Example Project
b. Understanding Filters
c. Working with Action Filters
i. Creating an Action Filter by Implementing IActionFilter
ii. Using the Convenience Action Filter Base Class
iii. Creating a Short-Circuiting Action Filter
d. Understanding the Filter Pipeline
i. Displaying the Filter Pipeline
ii. Understanding Filter Scope
e. Working with Authentication Filters
i. Preparing for Authentication
ii. Understanding Authentication Filters
iii. Creating an Authentication Filter
iv. Viewing the Filter Pipeline
3. Filters Part II
a. Preparing the Example Project
b. Reviewing Filters in the Dispatch Process
c. Working with Authorization Filters
i. Understanding Authorization Filters
ii. Creating an Authorization Filter
iii. Appling the Authorization Filter
iv. Removing the Authentication Filter
v. Using the Built-in Authorization Filter Attributes
vi. Reworking the Authentication Filter
d. Working with Exception Filters
i. Understanding the Default Behavior
ii. Understanding Exception Filters
iii. Creating an Exception Filter
iv. Deriving the Filter from the ExceptionFilterAttribute Class
e. Working with Override Filters
i. Overriding Built-in Filter Types
ii. Redefining Filter Policies

EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Understand controllers, ApiController dispatch process, filters, filter pipeline
• Able to customize the controller Dispatch Process
• Able to work with action filters, authentication filters, exception filters, override filters
• Able to customize URL routing
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.
</td>
<td>23/12/2024</td>
<td>16/01/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>133</td>
<td>Module08</td>
<td>STC CS MVC 5 Web API-808</td>
<td>
Error Handling and Using OWIN
1. Error Handling
a. Preparing the Example Project
b. Dealing with Errors
i. Relying on the Default Behavior
ii. Using an Implementation of the IHttpActionResult Interface
c. Using the HttpError Class
i. Using an Error Response and an HttpError Object
ii. Adding Extra Information to the HttpError Object
iii. Including Model State Errors in the HTTP Response
iv. Controlling Error Detail
v. Displaying HttpError Information in the Client
d. Responding to Errors Globally
i. Handling Exceptions
ii. Logging Exceptions
2. Using OWIN
a. Understanding OWIN
b. Creating a Self-hosted Web API Application
i. Creating the Project
ii. Installing the Packages
iii. Creating the Model and Repository
iv. Defining the Configuration Classes
v. Creating the Web API Controller
vi. Testing the Self-hosted Web API Application
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand Error and OWIN
• Able to deal with errors, use the HttpError class
• Able to respond to errors globally
• Able to create a self-hosted web API application</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>24/12/2024</td>
<td>18/01/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>134</td>
<td>Module09</td>
<td>STC CS ASP.NET CORE-901</td>
<td>
Getting started with ASP.NET Core and Create Your first application
1. Getting started with ASP.NET Core
a. An introduction to ASP.NET Core
i. Using a web framework
ii. What is ASP.NET Core?
b. When to choose ASP.NET Core
i. What type of applications can you build?
ii. If you're new to .NET development
iii. If you're a .NET Framework developer creating a new application
iv. Converting an existing ASP.NET application to ASP.NET Core
c. How does ASP.NET Core work?
i. How does an HTTP web request work?
ii. How does ASP.NET Core process a request?
2. Your first application
a. A brief overview of an ASP.NET Core application
b. Creating your first ASP.NET Core application
i. Using a template to get started
ii. Building the application
c. Running the web application
d. Understanding the project layout
e. The .csproj project file: Defining your dependencies
f. The Program class: Building a web host
g. The Startup class: Configuring your application
i. Adding and configuring services
ii. Defining how requests are handled with middleware
h. Generating responses with Razor Pages
i. Generating HTML with Razor Pages
ii. Handling request logic with PageModels and handlers
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Understand ASP.NET Core
• Create basic web application using ASP.NET CORE</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>26/12/2024</td>
<td>19/01/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>135</td>
<td>Module09</td>
<td>STC CS ASP.NET CORE-902</td>
<td>
Handling requests with the middleware pipeline and Creating a website with Razor Pages
<ol type="1">
Handling requests with the middleware pipeline
<ol type="a">
What is middleware?
Combining middleware in a pipeline
<ol type="i">
Simple pipeline scenario 1: A holding page
Simple pipeline scenario 2: Handling static files
Simple pipeline scenario 3: A Razor Pages application

Handling errors using middleware
<ol type="i">
Viewing exceptions in development: DeveloperExceptionPage
Handling exceptions in production: ExceptionHandlerMiddleware
Handling other errors: StatusCodePagesMiddleware
Error handling middleware and Web APIs

Creating a website with Razor Pages
<ol type="a">
An introduction to Razor Pages
<ol type="i">
Exploring a typical Razor Page
The MVC design pattern
Applying the MVC design pattern to Razor Pages
Adding Razor Pages to your application

Razor Pages vs. MVC in ASP.NET Core
<ol type="i">
MVC controllers in ASP.NET Core
The benefits of Razor Pages
When to choose MVC controllers over Razor Pages

Razor Pages and page handlers
<ol type="i">
Accepting parameters to page handlers
Returning responses with ActionResults

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Handling requests with the middleware pipeline
Creating a website with Razor Pages</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>28/12/2024</td>
<td>20/01/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>136</td>
<td>Module09</td>
<td>STC CS ASP.NET CORE-903</td>
<td>
Routing and Model Binding
1. Mapping URLs to Razor Pages using routing
a. What is routing?
b. Routing in ASP.NET Core
i. Using endpoint routing in ASP.NET Core
ii. Convention-based routing vs. attribute routing
iii. Routing to Razor Pages
c. Customizing Razor Page route templates
i. Adding a segment to a Razor Page route template
ii. Replacing a Razor Page route template completely
d. Exploring the route template syntax
i. Using optional and default values
ii. Adding additional constraints to route parameters
iii. Matching arbitrary URLs with the catch-all parameter
e. Generating URLs from route parameters
i. Generating URLs for a Razor Page
ii. Generating URLs for an MVC controller
iii. Generating URLs with ActionResults
iv. Generating URLs from other parts of your application
f. Selecting a page handler to invoke
g. Customizing conventions with Razor Pages
2. The binding model: Retrieving and validating user input
a. Understanding the models in Razor Pages and MVC
b. From request to model: Making the request useful
i. Binding simple types
ii. Binding complex types
iii. Choosing a binding source
c. Handling user input with model validation
i. The need for validation
ii. Using DataAnnotations attributes for validation
iii. Validating on the server for safety
iv. Validating on the client for user experience
d. Organizing your binding models in Razor Pages
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Creating and modifying convention-based routing and attribute routing
• Generating URLs from route parameters
• Binding model
• Validating user input</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>29/12/2024</td>
<td>21/01/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>137</td><td>Module09</td><td>STC CS ASP.NET CORE-904</td><td>
Rendering HTML using Razor views and Building forms with Tag Helpers

Rendering HTML using Razor views
<ol type="a">
Views: Rendering the user interface
Creating Razor views
<ol type="i">
Razor views and code-behind
Introducing Razor templates
Passing data to views

Creating dynamic web pages with Razor
<ol type="i">
Using C# in Razor templates
Adding loops and conditionals to Razor templates
Rendering HTML with Raw

Layouts, partial views, and _ViewStart
<ol type="i">
Using layouts for shared markup
Overriding parent layouts using sections
sing partial views to encapsulate markup
Running code on every view with _ViewStart and _ViewImports

Selecting a view from an MVC controller

Building forms with Tag Helpers
<ol type="a">
Catering to editors with Tag Helpers
Creating forms using Tag Helpers
<ol type="i">
The Form Tag Helper
The Label Tag Helper
The Input and Textarea Tag Helpers
The Select Tag Helper
The Validation Message and Validation Summary Tag Helpers

Generating links with the Anchor Tag Helper
Cache-busting with the Append Version Tag Helper
Using conditional markup with the Environment Tag Helper

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Render HTML using Razor views
Build forms with Tag Helpers</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>30/12/2024</td><td>22/01/2025</td></tr>
<tr><td>138</td><td>Module09</td><td>STC CS ASP.NET CORE-905</td><td>
Creating a Web API for mobile and client applications using MVC

What is a Web API and when should you use one?
Creating your first Web API project
Applying the MVC design pattern to a Web API
Attribute routing: Linking action methods to URLs
<ol type="a">
Combining route attributes to keep your route templates DRY
Using token replacement to reduce duplication in attribute routing
Handling HTTP verbs with attribute routing

Using common conventions with the [ApiController] attribute
Generating a response from a model
<ol type="a">
Customizing the default formatters: Adding XML support
Choosing a response format with content negotiation

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
create MVC web API
apply attribute routing into MVC web API
use conventions with the [ApiController] attribute
generate a response from a model</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>31/12/2024</td><td>23/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>139</td><td>Module09</td><td>STC CS ASP.NET CORE-906</td><td>Service configuration with dependency injection
1. Introduction to dependency injection
a. Understanding the benefits of dependency injection
b. Creating loosely coupled code
c. Dependency injection in ASP.NET Core
2. Using the dependency injection container
a. Adding ASP.NET Core framework services to the container
b. Registering your own services with the container
c. Registering services using objects and lambdas
d. Registering a service in the container multiple times
e. Injecting services into action methods, page handlers, and views
3. Understanding lifetimes: When are services created?
a. Transient: Everyone is unique
b. Scoped: Let's stick together
c. Singleton: There can be only one
d. Keeping an eye out for captured dependencies
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create and implement DI
• use DI containers
• understand and properly handle lifetimes</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>01/01/2025</td><td>25/01/2025</td></tr>
<tr><td>140</td><td>Module09</td><td>STC CS ASP.NET CORE-907</td><td>Configuring an ASP.NET Core application
1. Introducing the ASP.NET Core configuration model
2. Configuring your application with CreateDefaultBuilder
3. Building a configuration object for your app
a. Adding a configuration provider in Program.cs
b. Using multiple providers to override configuration values
c. Storing configuration secrets safely
d. Reloading configuration values when they change
4. Using strongly typed settings with the options pattern
a. Introducing the IOptions interface
b. Reloading strongly typed options with IOptionsSnapshot
c. Designing your options classes for automatic binding
d. Binding strongly typed settings without the IOptions interface
5. Configuring an application for multiple environments
a. Identifying the hosting environment
b. Loading environmentspecific configuration files
c. Setting the hosting environment
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• configure an ASP.NET Core application</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>02/01/2025</td><td>26/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>141</td><td>Module09</td><td>STC CS ASP.NET CORE-908</td><td>Saving data with Entity Framework Core
1. Introducing Entity Framework Core
a. What is EF Core?
b. Why use an object-relational mapper?
c. When should you choose EF Core?
d. Mapping a database to your application code
2. Adding EF Core to an application
a. Choosing a database provider and installing EF Core
b. Building a data model
c. Registering a data context
3. Managing changes with migrations
a. Creating your first migration
b. Adding a second migration
4. Querying data from and saving data to the database
a. Creating a record
b. Loading a list of records
c. Loading a single record
d. Updating a model with changes
5. Using EF Core in production applications
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• add EF Core to an application
• manage changes with migrations
• query data from and save data to the database
• use EF Core in production applications</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>04/01/2025</td><td>27/01/2025</td></tr>
<tr><td>142</td><td>Module09</td><td>STC CS ASP.NET CORE-909</td><td>The MVC and Razor Pages filter pipeline
1. Understanding filters and when to use them
a. The MVC filter pipeline
b. The Razor Pages filter pipeline
c. Filters or middleware: Which should you choose?
d. Creating a simple filter
e. Adding filters to your actions, controllers, Razor Pages, and globally
f. Understanding the order of filter execution
2. Creating custom filters for your application
a. Authorization filters: Protecting your APIs
b. Resource filters: Short-circuiting your action methods
c. Action filters: Customizing model binding and action results
d. Exception filters: Custom exception handling for your action methods
e. Result filters: Customizing action results before they execute
f. Page filters: Customizing model binding for Razor Pages
3. Understanding pipeline short-circuiting
4. Using dependency injection with filter attributes
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• use filters
• create custom filters
• use pipeline short-circuit
• use dependency injection with filter attributes</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>05/01/2025</td><td>28/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>143</td><td>Module09</td><td>STC CS ASP.NET CORE-910</td><td>
Authentication and Authorization
1. Authentication: Adding users to your application with Identity
a. Introducing authentication and authorization
i. Understanding users and claims in ASP.NET Core
ii. Authentication in ASP.NET Core: Services and middleware
iii. Authentication for APIs and distributed applications
b. What is ASP.NET Core Identity?
c. Creating a project that uses ASP.NET Core Identity
i. Creating the project from a template
ii. Exploring the template in Solution Explorer
iii. The ASP.NET Core Identity data model
iv. Interacting with ASP.NET Core Identity
d. Adding ASP.NET Core Identity to an existing project
i. Configuring the ASP.NET Core Identity services and middleware
ii. Updating the EF Core data model to support Identity
iii. Updating the Razor views to link to the Identity UI
e. Customizing a page in ASP.NET Core Identity's default UI
f. Managing users: Adding custom data to users
2. Authorization: Securing your application
a. Introduction to authorization
b. Authorization in ASP.NET Core
i. Preventing anonymous users from accessing your application
ii. Handling unauthorized requests
c. Using policies for claims-based authorization
d. Creating custom policies for authorization
i. Requirements and handlers: The building blocks of a policy
ii. Creating a policy with a custom requirement and handler
e. Controlling access with resource-based authorization
i. Manually authorizing requests with IAuthorizationService
ii. Creating a resource-based AuthorizationHandler
f. Hiding elements in Razor templates from unauthorized users
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create and use authorization and authentication</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>06/01/2025</td><td>29/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>144</td><td>Module09</td><td>STC CS ASP.NET CORE-911</td><td>
Publishing & deploying app, Monitoring & troubleshooting errors with logging, Improving app's security

Publishing and deploying your application

a. Understanding the ASP.NET Core hosting model

i. Running vs. publishing an ASP.NET Core app
ii. Choosing a deployment method for your application

b. Publishing your app to IIS

i. Configuring IIS for ASP.NET Core
ii. Preparing and publishing your application to IIS

c. Hosting an application on Linux

i. Running an ASP.NET Core app behind a reverse proxy on Linux
ii. Preparing your app for deployment to Linux

d. Configuring the URLs for your application
e. Optimizing your client-side assets using BundlerMinifier

i. Speeding up an app using bundling and minification
ii. Adding BundlerMinifier to your application
iii. Using minified files in production with the Environment Tag Helper
iv. Serving common files from a CDN

Monitoring and troubleshooting errors with logging

a. Using logging effectively in a production app

i. Highlighting problems using custom log messages
ii. The ASP.NET Core logging abstractions

b. Adding log messages to your application

i. Log level: How important is the log message?
ii. Log category: Which component created the log
iii. Formatting messages and capturing parameter values

c. Controlling where logs are written using logging providers

i. Adding a new logging provider to your application
ii. Replacing the default ILoggerFactory with Serilog

d. Changing log verbosity with filtering
e. Structured logging: Creating searchable, useful logs

i. Adding a structured logging provider to your app
ii. Using scopes to add additional properties to your logs

Improving your application's security

a. Adding HTTPS to an application

i. Using the ASP.NET Core HTTPS development certificates
ii. Configuring Kestrel with a production HTTPS certificate
iii. Enforcing HTTPS for your whole app

b. Defending against cross-site scripting (XSS) attacks
c. Protecting from cross-site request forgery (CSRF) attacks
d. Calling your web APIs from other domains using CORS

i. Understanding CORS and how it works
ii. Adding a global CORS policy to your whole app
iii. Adding CORS to specific Web API actions with EnableCorsAttribute
iv. Configuring CORS policies

e. Exploring other attack vectors

i. Detecting and avoiding open redirect attacks
ii. Avoiding SQL injection attacks with EF Core and parameterization
iii. Preventing insecure direct object references
iv. Protecting your users' passwords and data

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
publish & deploy application
monitor & troubleshoot an application errors with logging
improve application's security</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>07/01/2025</td><td>30/01/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>145</td><td>Module09</td><td>STC CS ASP.NET CORE-912</td><td>
Building custom components
1. Building custom components
a. Customizing your middleware pipeline
i. Creating simple endpoints with the Run extension
ii. Branching middleware pipelines with the Map extension
iii. Adding to the pipeline with the Use extension
iv. Building a custom middleware component
b. Creating custom endpoints with endpoint routing
i. Creating a custom endpoint routing component
ii. Creating simple endpoints with MapGet and WriteJsonAsync
iii. Applying authorization to endpoints
c. Handling complex configuration requirements
i. Partially building configuration to configure additional providers
ii. Using services to configure IOptions with IConfigureOptions
d. Using a third-party dependency injection container
2. Building custom MVC and Razor Pages components
a. Creating a custom Razor Tag Helper
i. Printing environment information with a custom Tag Helper
ii. Creating a custom Tag Helper to conditionally hide elements
iii. Creating a Tag Helper to convert Markdown to HTML
b. View components: Adding logic to partial views
c. Building a custom validation attribute
d. Replacing the validation framework with FluentValidation
i. Comparing FluentValidation to DataAnnotations attributes
ii. Adding FluentValidation to your application
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• build custom components
• build custom MVC and Razor Pages components</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>08/01/2025</td><td>01/02/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>146</td><td>Module09</td><td>STC CS ASP.NET CORE-912</td><td>Building custom components
1. Building custom components
a. Customizing your middleware pipeline
i. Creating simple endpoints with the Run extension
ii. Branching middleware pipelines with the Map extension
iii. Adding to the pipeline with the Use extension
iv. Building a custom middleware component
b. Creating custom endpoints with endpoint routing
i. Creating a custom endpoint routing component
ii. Creating simple endpoints with MapGet and WriteJsonAsync
iii. Applying authorization to endpoints
c. Handling complex configuration requirements
i. Partially building configuration to configure additional providers
ii. Using services to configure IOptions with IConfigureOptions
d. Using a third-party dependency injection container
2. Building custom MVC and Razor Pages components
a. Creating a custom Razor Tag Helper
i. Printing environment information with a custom Tag Helper
ii. Creating a custom Tag Helper to conditionally hide elements
iii. Creating a Tag Helper to convert Markdown to HTML
b. View components: Adding logic to partial views
c. Building a custom validation attribute
d. Replacing the validation framework with FluentValidation
i. Comparing FluentValidation to DataAnnotations attributes
ii. Adding FluentValidation to your application
<table><tr><th colspan="2">EVIDENCE GUIDANCE</th></tr><tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• build custom components
• build custom MVC and Razor Pages components</td></tr><tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr></table></td><td>09/01/2025</td><td>02/02/2025</td></tr>
<tr><td>0</td><td></td><td></td><td>Exam Review Class</td><td>11/01/2025</td><td></td></tr>
<tr><td>0</td><td></td><td></td><td>Exam</td><td>12/01/2025</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>147</td>
<td>Module09</td>
<td>STC CS ASP.NET CORE-913</td>
<td>
Calling remote APIs, Building background tasks and services, Testing application
<ol type="1">
Calling remote APIs with IHttpClientFactory
<ol type="a">
Calling HTTP APIs: The problem with HttpClient
Creating HttpClients with IHttpClientFactory
<ol type="i">
Using IHttpClientFactory to manage HttpClientHandler lifetime
Configuring named clients at registration time
Using typed clients to encapsulate HTTP calls

Handling transient HTTP errors with Polly
Creating a custom HttpMessageHandler

Building background tasks and services
<ol type="a">
Running background tasks with IHostedService
<ol type="i">
Running background tasks on a timer
Using scoped services in background tasks

Creating headless worker services using IHost
<ol type="i">
Creating a worker service from a template
Running worker services in production

Coordinating background tasks using Quartz.NET
<ol type="i">
Installing Quartz.NET in an ASP.NET Core application
Configuring a job to run on a schedule with Quartz.NET
Using clustering to add redundancy to your background tasks

Testing your application
<ol type="a">
An introduction to testing in ASP.NET Core
Unit testing with xUnit
<ol type="i">
Creating your first test project
Running tests with dotnet test
Referencing your app from your test project
Adding Fact and Theory unit tests
Testing failure conditions

Unit testing custom middleware
Unit testing API controllers
Integration testing: Testing your whole app in-memory
<ol type="i">
Creating a TestServer using the Test Host package
Testing your application with WebApplicationFactory
Replacing dependencies in WebApplicationFactory
Reducing duplication by creating a custom WebApplicationFactory

Isolating the database with an in-memory EF Core provider

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
call remote APIs with IHttpClientFactory
build background tasks and services
test application</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>13/01/2025</td>
<td>03/02/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>148</td>
<td>Module09</td>
<td>STC CS ASP.NET CORE-913</td>
<td>
Calling remote APIs, Building background tasks and services, Testing application
<ol type="1">
Calling remote APIs with IHttpClientFactory
<ol type="a">
Calling HTTP APIs: The problem with HttpClient
Creating HttpClients with IHttpClientFactory
<ol type="i">
Using IHttpClientFactory to manage HttpClientHandler lifetime
Configuring named clients at registration time
Using typed clients to encapsulate HTTP calls

Handling transient HTTP errors with Polly
Creating a custom HttpMessageHandler

Building background tasks and services
<ol type="a">
Running background tasks with IHostedService
<ol type="i">
Running background tasks on a timer
Using scoped services in background tasks

Creating headless worker services using IHost
<ol type="i">
Creating a worker service from a template
Running worker services in production

Coordinating background tasks using Quartz.NET
<ol type="i">
Installing Quartz.NET in an ASP.NET Core application
Configuring a job to run on a schedule with Quartz.NET
Using clustering to add redundancy to your background tasks

Testing your application
<ol type="a">
An introduction to testing in ASP.NET Core
Unit testing with xUnit
<ol type="i">
Creating your first test project
Running tests with dotnet test
Referencing your app from your test project
Adding Fact and Theory unit tests
Testing failure conditions

Unit testing custom middleware
Unit testing API controllers
Integration testing: Testing your whole app in-memory
<ol type="i">
Creating a TestServer using the Test Host package
Testing your application with WebApplicationFactory
Replacing dependencies in WebApplicationFactory
Reducing duplication by creating a custom WebApplicationFactory

Isolating the database with an in-memory EF Core provider

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
call remote APIs with IHttpClientFactory
build background tasks and services
test application</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>14/01/2025</td>
<td>04/02/2025</td>
</tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
149	Module10	STC CS EF Core-1001	**Creating Entities, Database from Code and Seeding Data** 1. Installation a. Install Visual Studio b. Create a Project c. Install Entity Framework d. Install the Core Tools Package e. SQL Server Database f. Summary 2. Project Structure a. Core Project b. Unit Testing Overview c. Data Access Layer Project d. Main Project References e. Summary 3. Entities a. Person Entity b. Address Entity c. Simple Navigation Property d. Entity Property Constraints e. Entity Schema Attributes f. More Entity Schema Attributes g. Mapping of the Primary Key Column h. Enum Mapping i. Entity Inheritance Mapping i. Table-per-Hierarchy ii. Table-per-Type j. Summary 4. Database Context a. Creating a Simple Database Context b. Connecting to Our Database c. Accessing Entities in a Database Context d. Saving Entity Changes e. Configuring a Database Context i. Set a Database Schema for All Entities ii. Composite Key Constraint iii. Primary Key Constraint iv. Default Property Value v. Foreign Key Relationships f. Summary 5. Creating a Database from Code a. Setting the Connection String b. Creating the Initial Migration c. Creating the Database from the Migration d. Connect to the New Database from Visual Studio e. Summary 6. Seeding Data a. Populating Lookup Data b. Seeding Test Persons c. Seeding Addresses d. Summary	15/01/2025	05/02/2025

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
150	Module10	STC CS EF Core-1002	**Getting, Inserting, Updating, Deleting Data and Navigation Properties** 1. Getting Data a. LINQ Queries i. Select Anonymous Types ii. Select Object Transformation iii. Joins in LINQ iv. Select the Navigation Property v. Sorting Data b. Method Syntax Queries c. Testing Our Data d. Summary 2. Inserting Data a. Inserting the Root Entity b. Inserting Child Records c. Primary Key Values i. Identity Seeded Primary Key ii. Guid Primary Key iii. Non-computed Primary Key d. Foreign Key Values e. Default Values f. Record Insertion Integration Tests g. Summary 3. Updating Data a. Updating the Root Entity b. Updating a Child Entity c. Integration Test d. Summary 4. Deleting Data a. Deleting the Root Entity b. Deleting a Child Entity i. Cascade Delete ii. Client Set Null Delete Behavior iii. Restrict Delete Behavior iv. Set Null Delete Behavior c. Integration Test d. Summary 5. Navigation Properties a. Mapping a Parent Entity in a One-to-Many Relationship b. Mapping a Related Entity to Parent c. One-to-Many Integration Test d. Many-to-Many Relationships i. Map a Relationship Through Navigation Properties ii. Create and Run Migration iii. Many-to-Many Integration Test e. Summary	16/01/2025	06/02/2025

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
151	Module10	STC CS EF Core-1003	**Aggregations, Stored Procedures and Migrations** 1. Aggregations a. Unit Test Setup b. Count c. Adding Age to Person d. Min e. Max f. Average i. Average Unit Test g. Sum h. Group By i. Direct Group By ii. Group By with Count iii. Group By with Min iv. Group By with Max v. Group By with Average vi. Group By with Sum i. Summary 2. Stored Procedures a. Add Stored Procedures to a Database b. Set Up Unit Tests c. Test the GetPersonsByState Stored Procedure d. Test the AddLookUpItem Stored Procedure e. Summary 3. Migrations a. What Is a Migration? i. How to Add a Migration? ii. The Migration API b. Common Schema Commands i. Add a Column to a Table ii. Change a Column in a Table c. Common Data Commands i. Inserting Data ii. Deleting Data iii. Update Data in a Migration d. How to Run Migrations i. Update to the Most Recent Migration ii. Migrate to a Specific Migration iii. Run Migration from Code e. Automatic Data Migrations f. Summary	18/01/2025	10/02/2025

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
152	Module10	STC CS EF Core-1003	**Aggregations, Stored Procedures and Migrations** 1. Aggregations a. Unit Test Setup b. Count c. Adding Age to Person d. Min e. Max f. Average i. Average Unit Test g. Sum h. Group By i. Direct Group By ii. Group By with Count iii. Group By with Min iv. Group By with Max v. Group By with Average vi. Group By with Sum i. Summary 2. Stored Procedures a. Add Stored Procedures to a Database b. Set Up Unit Tests c. Test the GetPersonsByState Stored Procedure d. Test the AddLookUpItem Stored Procedure e. Summary 3. Migrations a. What Is a Migration? i. How to Add a Migration? ii. The Migration API b. Common Schema Commands i. Add a Column to a Table ii. Change a Column in a Table c. Common Data Commands i. Inserting Data ii. Deleting Data iii. Update Data in a Migration d. How to Run Migrations i. Update to the Most Recent Migration ii. Migrate to a Specific Migration iii. Run Migration from Code e. Automatic Data Migrations f. Summary	19/01/2025	11/02/2025

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>153</td>
<td>Module11</td>
<td>STC CS CORE Web API-1101</td>
<td>
Getting started
<ol type="1">
Web APIs at a glance
<ol type="a">
Web APIs
<ol type="i">
Overview
Real-world example
Types of web APIs
Architectures and message protocols

ASP.NET Core
<ol type="i">
Architecture
Program.cs
Controllers
Minimal APIs
Task-based asynchronous pattern

Our first web API project
<ol type="a">
System requirements
<ol type="i">
.NET SDK
Integrated development environment

Installing Visual Studio
Creating the web API project
MyBGList project overview
<ol type="i">
Reviewing launchSettings.json
Configuring the appsettings.json
Playing with the Program.cs file
Inspecting the WeatherForecastController
Adding the BoardGameController

Exercises
<ol type="i">
launchSettings.json
appsettings.json
Program.cs
BoardGame.cs
BoardGameControllers.cs

RESTful principles and guidelines
<ol type="a">
REST guiding constraints
<ol type="i">
Client-server approach
Statelessness
Cacheability
Layered system
Code on demand
Uniform interface

API documentation
<ol type="i">
Introducing OpenAPI
ASP.NET Core components

API versioning
<ol type="i">
Understanding versioning
Should we really use versions?
Implementing versioning

Exercises
<ol type="i">
CORS
Client-side caching
COD
API documentation and versioning

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
create ASP.NET CORE web API
implement RESTful principles</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>20/01/2025</td>
<td>12/02/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>154</td>
<td>Module11</td>
<td>STC CS CORE Web API-1101</td>
<td>
Getting started
<ol type="1">
Web APIs at a glance
<ol type="a">
Web APIs
<ol type="i">
Overview
Real-world example
Types of web APIs
Architectures and message protocols

ASP.NET Core
<ol type="i">
Architecture
Program.cs
Controllers
Minimal APIs
Task-based asynchronous pattern

Our first web API project
<ol type="a">
System requirements
<ol type="i">
.NET SDK
Integrated development environment

Installing Visual Studio
Creating the web API project
MyBGList project overview
<ol type="i">
Reviewing launchSettings.json
Configuring the appsettings.json
Playing with the Program.cs file
Inspecting the WeatherForecastController
Adding the BoardGameController

Exercises
<ol type="i">
launchSettings.json
appsettings.json
Program.cs
BoardGame.cs
BoardGameControllers.cs

RESTful principles and guidelines
<ol type="a">
REST guiding constraints
<ol type="i">
Client-server approach
Statelessness
Cacheability
Layered system
Code on demand
Uniform interface

API documentation
<ol type="i">
Introducing OpenAPI
ASP.NET Core components

API versioning
<ol type="i">
Understanding versioning
Should we really use versions?
Implementing versioning

Exercises
<ol type="i">
CORS
Client-side caching
COD
API documentation and versioning

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
create ASP.NET CORE web API
implement RESTful principles</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>21/01/2025</td>
<td>13/02/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>155</td>
<td>Module11</td>
<td>STC CS CORE Web API-1101</td>
<td>
Getting started
1. Web APIs at a glance
a. Web APIs
i. Overview
ii. Real-world example
iii. Types of web APIs
iv. Architectures and message protocols
b. ASP.NET Core
i. Architecture
ii. Program.cs
iii. Controllers
iv. Minimal APIs
v. Task-based asynchronous pattern
2. Our first web API project
a. System requirements
i. .NET SDK
ii. Integrated development environment
b. Installing Visual Studio
c. Creating the web API project
d. MyBGList project overview
i. Reviewing launchSettings.json
ii. Configuring the appsettings.json
iii. Playing with the Program.cs file
iv. Inspecting the WeatherForecastController
v. Adding the BoardGameController
e. Exercises
i. launchSettings.json
ii. appsettings.json
iii. Program.cs
iv. BoardGame.cs
v. BoardGameControllers.cs
3. RESTful principles and guidelines
a. REST guiding constraints
i. Client-server approach
ii. Statelessness
iii. Cacheability
iv. Layered system
v. Code on demand
vi. Uniform interface
b. API documentation
i. Introducing OpenAPI
ii. ASP.NET Core components
c. API versioning
i. Understanding versioning
ii. Should we really use versions?
iii. Implementing versioning
d. Exercises
i. CORS
ii. Client-side caching
iii. COD
iv. API documentation and versioning
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create ASP.NET CORE web API
• implement RESTful principles</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>22/01/2025</td>
<td>16/02/2025</td>
</tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
156	Module11	STC CS CORE Web API-1102	**Basic concepts** 1. Working with data a. Choosing a database i. Comparing SQL and NoSQL ii. Making a choice b. Creating the database i. Obtaining the CSV file ii. Installing SQL Server iii. Installing SSMS or ADS iv. Adding a new database c. EF Core i. Reasons to use an ORM ii. Setting up EF Core iii. Creating the DbContext iv. Setting up the DbContext v. Creating the database structure d. Exercises i. Additional fields ii. One-to-many relationship iii. Many-to-many relationship iv. Creating a new migration v. Applying the new migration vi. Reverting to a previous migration 2. CRUD operations a. Introducing LINQ i. Query syntax vs. method syntax ii. Lambda expressions iii. The IQueryable<t> interface b. Injecting the DbContext i. The sync and async methods ii. Testing the ApplicationDbContext c. Seeding the database i. Setting up the CSV file ii. Installing the CsvHelper package iii. Creating the BggRecord class iv. Adding the SeedController v. Reading the CSV file vi. Executing the SeedController d. Reading data i. Paging ii. Sorting iii. Filtering e. Updating and deleting data i. Updating a BoardGame ii. Deleting a BoardGame f. Exercises i. Create ii. Read iii. Update iv. Delete 3. Data validation and error handling a. Data validation i. Model binding ii. Data validation attributes iii. A nontrivial validation example iv. Data validation and OpenAPI v. Binding complex types b. Error handling i. The ModelState object ii. Custom error messages iii. Manual model validation iv. Exception handling c. Exercises i. Built-in validators ii. Custom validators iii. IValidatableObject iv. ModelState validation v. Exception handling **EVIDENCE GUIDANCE**	23/01/2025	17/02/2025

Sl	Module	Competency Unit	Elements		Schedule Date	Actual Date
			Critical aspects of evidence	**Assessment must confirm the ability to** – • work with data • apply CRUD operations • check data validation • handle error		
			Assessment of unit	This unit may be assessed by taking **minimum one (1) quiz** after completion of the unit. Each quiz will contain **10 questions of 1 mark each**. A **total mark of each quiz is 10**. **Quiz time is 10 minutes**.		

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
157	Module11	STC CS CORE Web API-1102	**Basic concepts** 1. Working with data a. Choosing a database i. Comparing SQL and NoSQL ii. Making a choice b. Creating the database i. Obtaining the CSV file ii. Installing SQL Server iii. Installing SSMS or ADS iv. Adding a new database c. EF Core i. Reasons to use an ORM ii. Setting up EF Core iii. Creating the DbContext iv. Setting up the DbContext v. Creating the database structure d. Exercises i. Additional fields ii. One-to-many relationship iii. Many-to-many relationship iv. Creating a new migration v. Applying the new migration vi. Reverting to a previous migration 2. CRUD operations a. Introducing LINQ i. Query syntax vs. method syntax ii. Lambda expressions iii. The IQueryable<t> interface b. Injecting the DbContext i. The sync and async methods ii. Testing the ApplicationDbContext c. Seeding the database i. Setting up the CSV file ii. Installing the CsvHelper package iii. Creating the BggRecord class iv. Adding the SeedController v. Reading the CSV file vi. Executing the SeedController d. Reading data i. Paging ii. Sorting iii. Filtering e. Updating and deleting data i. Updating a BoardGame ii. Deleting a BoardGame f. Exercises i. Create ii. Read iii. Update iv. Delete 3. Data validation and error handling a. Data validation i. Model binding ii. Data validation attributes iii. A nontrivial validation example iv. Data validation and OpenAPI v. Binding complex types b. Error handling i. The ModelState object ii. Custom error messages iii. Manual model validation iv. Exception handling c. Exercises i. Built-in validators ii. Custom validators iii. IValidatableObject iv. ModelState validation v. Exception handling **EVIDENCE GUIDANCE**	25/01/2025	18/02/2025

Sl	Module	Competency Unit	Elements		Schedule Date	Actual Date
			Critical aspects of evidence	**Assessment must confirm the ability to** – • work with data • apply CRUD operations • check data validation • handle error		
			Assessment of unit	This unit may be assessed by taking **minimum one (1) quiz** after completion of the unit. Each quiz will contain **10 questions of 1 mark each**. A **total mark of each quiz is 10**. **Quiz time is 10 minutes**.		

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>158</td><td>Module11</td><td>STC CS CORE Web API-1103</td><td>
Application logging and Caching techniques
<ol type="1">
Application logging
<ol type="a">
Application logging overview
<ol type="i">
From boats to computers
Why do we need logs?
ASP.NET logging
<ol type="i">
A quick logging test
Log levels
Logging configuration
Logging providers
Event IDs and templates
Exception logging
Unstructured vs. structured logging
<ol type="i">
Unstructured logging pros and cons
Structured logging advantages
Application Insights logging provider
Third-party logging providers
<ol type="i">
Serilog overview
Installing Serilog
Configuring Serilog
Testing Serilog
Improving the logging behavior
Exercises
<ol type="i">
JSON console logging
Logging provider configuration
Exception logging's new property
New Serilog enricher
New Serilog sink

Caching techniques
<ol type="a">
Caching overview
HTTP response caching
<ol type="i">
Setting the cache-control header manually
Adding a default caching directive
Defining cache profiles
Server-side response caching
Response caching vs. client reload
In-memory caching
<ol type="i">
Setting up the in-memory cache
Injecting the IMemoryCache interface
Using the in-memory cache
Distributed caching
<ol type="i">
Distributed cache providers overview
SQL Server
Redis
Exercises
<ol type="i">
HTTP response caching
Cache profiles
Server-side response caching
In-memory caching
Distributed caching

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
application log
manage & handle cache</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>26/01/2025</td><td>19/02/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>159</td>
<td>Module11</td>
<td>STC CS CORE Web API-1104</td>
<td>
Authentication and authorization

Basic concepts
<ol type="a">
Authentication
Authorization

ASP.NET Core Identity
<ol type="a">
Installing the NuGet packages
Creating the user entity
Updating the ApplicationDbContext
Adding and applying a new migration
Setting up the services and middleware
Implementing the AccountController

Authorization settings
<ol type="a">
Adding the authorization HTTP header
Setting up the [authorize] attribute
Testing the authorization flow

Role-based access control
<ol type="a">
Registering new users
Creating the new roles
Assigning users to roles
Adding role-based claims to JWT
Setting up role-based auth rules
Testing the RBAC flow
Using alternative authorization methods

Exercises
<ol type="a">
Adding a new role
Creating a new user
Assigning a user to roles
Implementing a test endpoint
Testing the RBAC flow

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
authenticate and authorize user</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>27/01/2025</td>
<td>20/02/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>160</td>
<td>Module11</td>
<td>STC CS CORE Web API-1105</td>
<td>
Beyond REST

REST drawbacks
<ol type="a">
Overfetching
Underfetching

GraphQL
<ol type="a">
GraphQL advantages
GraphQL drawbacks
Implementing GraphQL
Working with GraphQL

Google Remote Procedure Call
<ol type="a">
gRPC pros
gRPC cons
Installing the NuGet packages
Implementing the gRPC Server
Implementing the gRPC client
Adding Authorization support

Other REST alternatives
<ol type="a">
Newline Delimited JSON (NDJSON)
Falcor
Thrift

Exercises
<ol type="a">
Write a new GraphQL query
Fetch GraphQL data for a mutation
Implement new gRPC server features
Add new gRPC client wrappers
Test the new gRPC features

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
implement GraphQL and work with it
implement gRPC server and gRPC client
add authorization support to gRPC
implement other REST alternatives: Newline Delimited JSON (NDJSON), Falcor & Thrift</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>28/01/2025</td>
<td>22/02/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>161</td><td>Module11</td><td>STC CS CORE Web API-1106</td><td>
Toward production
1. API documentation
a. Web API potential audience
i. Prospectors
ii. Contractors
iii. Builders
b. API documentation best practices
i. Adopt an automated description tool
ii. Describe endpoints and input parameters
iii. Add XML documentation support
iv. Work with Swashbuckle annotations
v. Describe responses
vi. Add request and response samples
vii. Group endpoints into sections
viii. Exclude reserved endpoints
c. Filter-based Swagger customization
i. Emphasizing the authorization requirements
ii. Changing the application title
iii. Adding a warning text for passwords
iv. Adding custom key/value pairs
d. Exercises
i. Use XML documentation
ii. Use Swashbuckle annotations
iii. Exclude some endpoints
iv. Add a custom filter
v. Add custom key/value pairs
2. Release and deployment
a. Prepublishing tasks
i. Considering security
ii. Choosing a domain name
iii. Setting up a CDN
iv. Fine-tuning our APP
v. Understanding the .NET publishing modes
b. Creating a Windows VM server
i. Accessing Azure
ii. Creating and setting up the Windows VM
iii. Working with the VM public IP address
iv. Creating an SSL/TLS origin certificate
v. Setting Cloudflare Encryption Mode to Full
c. Configuring the Windows VM server
i. Installing IIS
ii. Installing the ASP.NET Core hosting bundle
iii. Installing the Web Deploy component
iv. Opening the 8172 TCP port
v. Configuring IIS
vi. Creating the production database
vii. Creating the appsettings.Production.json file
d. Publishing and deploying
i. Introducing Visual Studio publish profiles
ii. Creating an Azure VM publish profile
iii. Configuring the publish profile
iv. Publishing, deployment, and testing
v. Final thoughts
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create API document
• release and deploy web API</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>29/01/2025</td><td>23/02/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>162</td>
<td>Module12</td>
<td>STC CS Angular-1201</td>
<td>
Getting Started with Angular
1. Getting Ready
a. What Do You Need to Know?
b. What Is the Structure of This Book?
i. Part 1: Getting Started with Angular
ii. Part 2: Angular in Detail
iii. Part 3: Advanced Angular Features
c. What Software Do I Need for Angular Development?
d. How Do I Set Up the Development Environment?
e. Are There Lots of Examples?
f. Where Can You Get the Example Code?
g. How Do You Set Up Your Development Environment?
2. Jumping Right In
a. Getting Ready
i. Installing Node.js
ii. Installing an Editor
iii. Installing the Angular Development Package
iv. Choosing a Browser
b. Creating and Preparing the Project
i. Opening the Project for Editing
ii. Starting the Angular Development Tools
c. Adding Features to the Application
i. Creating a Data Model
ii. Displaying Data to the User
iii. Updating the Component
d. Styling the Application Content
i. Applying Angular Material Components
ii. Defining the Spacer CSS Style
e. Displaying the List of To-Do Items
i. Defining Additional Styles
f. Creating a Two-Way Data Binding
g. Filtering Completed To-Do Items
h. Adding To-Do Items
i. Finishing Up
3. Primer, Part 1
a. Preparing the Example Project
b. Understanding HTML
i. Understanding Void Elements
ii. Understanding Attributes
iii. Applying Attributes Without Values
iv. Quoting Literal Values in Attributes
v. Understanding Element Content
vi. Understanding the Document Structure
c. Understanding CSS and the Bootstrap Framework
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
○ create Angular web application
○ combine angular with HTML and CSS
○ Applying Angular Material Components
○ Creating a Two-Way Data Binding
○ Filtering and Adding</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>30/01/2025</td>
<td>24/02/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>163</td><td>Module12</td><td>STC CS Angular-1202</td><td>
JavaScript and TypeScript
1. Primer, Part 1
a. Understanding TypeScript/JavaScript
i. Understanding the TypeScript Workflow
ii. Understanding JavaScript vs. TypeScript
iii. Understanding the Basic TypeScript/JavaScript Features
iv. Defining Variables and Constants
v. Dealing with Unassigned and Null Values
vi. Using the JavaScript Primitive Types
vii. Using the JavaScript Operators
2. Primer, Part 2
a. Preparing for This Chapter
b. Defining and Using Functions
i. Defining Optional Function Parameters
ii. Defining Default Parameter Values
iii. Defining Rest Parameters
iv. Defining Functions That Return Results
v. Using Functions as Arguments to Other Functions
c. Working with Arrays
i. Reading and Modifying the Contents of an Array
ii. Enumerating the Contents of an Array
iii. Using the Spread Operator
iv. Using the Built-in Array Methods
d. Working with Objects
i. Understanding Literal Object Types
ii. Defining Classes
iii. Checking Object Types
e. Working with JavaScript Modules
i. Creating and Using Modules
f. Working with Reactive Extensions
i. Understanding Observables
ii. Understanding Observers
iii. Understanding Subjects
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Work with JavaScript and TypeScript and Reactive Extensions</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>01/02/2025</td><td>25/02/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>164</td><td>Module12</td><td>STC CS Angular-1102</td><td>
JavaScript and TypeScript
1. Primer, Part 1
a. Understanding TypeScript/JavaScript
i. Understanding the TypeScript Workflow
ii. Understanding JavaScript vs. TypeScript
iii. Understanding the Basic TypeScript/JavaScript Features
iv. Defining Variables and Constants
v. Dealing with Unassigned and Null Values
vi. Using the JavaScript Primitive Types
vii. Using the JavaScript Operators
2. Primer, Part 2
a. Preparing for This Chapter
b. Defining and Using Functions
i. Defining Optional Function Parameters
ii. Defining Default Parameter Values
iii. Defining Rest Parameters
iv. Defining Functions That Return Results
v. Using Functions as Arguments to Other Functions
c. Working with Arrays
i. Reading and Modifying the Contents of an Array
ii. Enumerating the Contents of an Array
iii. Using the Spread Operator
iv. Using the Built-in Array Methods
d. Working with Objects
i. Understanding Literal Object Types
ii. Defining Classes
iii. Checking Object Types
e. Working with JavaScript Modules
i. Creating and Using Modules
f. Working with Reactive Extensions
i. Understanding Observables
ii. Understanding Observers
iii. Understanding Subjects
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Work with JavaScript and TypeScript and Reactive Extensions</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>02/02/2025</td><td>26/02/2025</td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
165	Module12	STC CS Angular-1203	**Sports Store: A Real Application** 1. SportsStore: A Real Application a. Preparing the Project i. Installing the Additional NPM Packages ii. Preparing the RESTful Web Service iii. Preparing the HTML File iv. Creating the Folder Structure v. Running the Example Application vi. Starting the RESTful Web Service b. Preparing the Angular Project Features i. Updating the Root Component ii. Updating the Root Module iii. Inspecting the Bootstrap File c. Starting the Data Model i. Creating the Model Classes ii. Creating the Dummy Data Source iii. Creating the Model Repository iv. Creating the Feature Module d. Starting the Store i. Creating the Store Component and Template ii. Creating the Store Feature Module iii. Updating the Root Component and Root Module e. Adding Store Features the Product Details i. Displaying the Product Details ii. Adding Category Selection iii. Adding Product Pagination iv. Creating a Custom Directive 2. Spor tsStore: Orders and Checkout a. Preparing the Example Application b. Creating the Cart i. Creating the Cart Model ii. Creating the Cart Summary Components iii. Integrating the Cart into the Store c. Adding URL Routing i. Creating the Cart Detail and Checkout Components ii. Creating and Applying the Routing Configuration iii. Navigating Through the Application iv. Guarding the Routes d. Completing the Cart Detail Feature e. Processing Orders i. Extending the Model ii. Collecting the Order Details f. Using the RESTful Web Service i. Applying the Data Source 3. SportsStore: Administration a. Preparing the Example Application i. Creating the Module ii. Configuring the URL Routing System iii. Navigating to the Administration URL b. Implementing Authentication i. Understanding the Authentication System ii. Extending the Data Source iii. Creating the Authentication Service iv. Enabling Authentication c. Extending the Data Source and Repositories d. Installing the Component Library e. Creating the Administration Feature Structure i. Creating the Placeholder Components ii. Preparing the Common Content and the Feature Module iii. Implementing the Product Table Feature iv. Implementing the Product Editor v. Implementing the Order Table Feature 4. SportsStore: Progressive Features and Deployment a. Preparing the Example Application b. Adding Progressive Features i. Installing the PWA Package ii. Caching the Data URLs iii. Responding to Connectivity Changes c. Preparing the Application for Deployment i. Creating the Data File ii. Creating the Server	03/02/2025	27/02/2025

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td></td><td></td><td></td><td>iii. Changing the Web Service URL in the Repository Class
d. Building and Testing the Application
i. Testing the Progressive Features
e. Containerizing the SportsStore Application
i. Installing Docker
ii. Preparing the Application
iii. Creating the Docker Container
iv. Running the Application

EVIDENCE GUIDANCE

Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• create angular web application

Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td></td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
166	Module12	STC CS Angular-1204	**Angular Data Bindings, Built-in Directives, Events and Forms** 1. Understanding Angular Projects and Tools a. Creating a New Angular Project b. Understanding the Project Structure i. Understanding the Source Code Folder ii. Understanding the Packages Folder c. Using the Development Tools i. Understanding the Development HTTP Server ii. Understanding the Build Process iii. Using the Linter d. Understanding How an Angular Application Works i. Understanding the HTML Document ii. Understanding the Application Bootstrap iii. Understanding the Root Angular Module iv. Understanding the Angular Component v. Understanding Content Display e. Understanding the Production Build Process i. Running the Production Build f. Starting Development in an Angular Project i. Creating the Data Model ii. Creating a Component and Template iii. Configuring the Root Angular Module 2. Using Data Bindings a. Preparing for This Chapter b. Understanding One-Way Data Bindings i. Understanding the Binding Target ii. Understanding the Expression iii. Understanding the Brackets iv. Understanding the Host Element c. Using the Standard Property and Attribute Bindings i. Using the Standard Property Binding ii. Using the String Interpolation Binding iii. Using the Attribute Binding d. Setting Classes and Styles i. Using the Class Bindings ii. Using the Style Bindings e. Updating the Data in the Application 3. Using the Built-in Directives a. Preparing the Example Project b. Using the Built-in Directives i. Using the ngIf Directive ii. Using the ngSwitch Directive iii. Using the ngFor Directive iv. Using the ngTemplateOutlet Directive v. Using Directives Without an HTML Element c. Understanding One-Way Data Binding Restrictions i. Using Idempotent Expressions ii. Understanding the Expression Context 4. Using Events and Forms a. Preparing the Example Project i. Importing the Forms Module ii. Preparing the Component and Template b. Using the Event Binding i. Using Event Data ii. Handling Events in the Component iii. Using Template Reference Variables c. Using Two-Way Data Bindings i. Using the ngModel Directive d. Working with Forms i. Adding a Form to the Example Application ii. Adding Form Data Validation iii. Validating the Entire Form iv. Completing the Form **EVIDENCE GUIDANCE** Critical aspects of evidence: **Assessment must confirm the ability to –** • **create angular web application using Data Bindings, Built-in Directives, Events and Forms**	04/02/2025	01/03/2025

Sl	Module	Competency Unit	Elements		Schedule Date	Actual Date
			Assessment of unit	This unit may be assessed by taking **minimum one (1) quiz** after completion of the unit. Each quiz will contain **10 questions of 1 mark each**. A **total mark of each quiz is 10**. **Quiz time is 10 minutes**.		

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
167	Module12	STC CS Angular-1204	**Angular Data Bindings, Built-in Directives, Events and Forms** 1. Understanding Angular Projects and Tools a. Creating a New Angular Project b. Understanding the Project Structure i. Understanding the Source Code Folder ii. Understanding the Packages Folder c. Using the Development Tools i. Understanding the Development HTTP Server ii. Understanding the Build Process iii. Using the Linter d. Understanding How an Angular Application Works i. Understanding the HTML Document ii. Understanding the Application Bootstrap iii. Understanding the Root Angular Module iv. Understanding the Angular Component v. Understanding Content Display e. Understanding the Production Build Process i. Running the Production Build f. Starting Development in an Angular Project i. Creating the Data Model ii. Creating a Component and Template iii. Configuring the Root Angular Module 2. Using Data Bindings a. Preparing for This Chapter b. Understanding One-Way Data Bindings i. Understanding the Binding Target ii. Understanding the Expression iii. Understanding the Brackets iv. Understanding the Host Element c. Using the Standard Property and Attribute Bindings i. Using the Standard Property Binding ii. Using the String Interpolation Binding iii. Using the Attribute Binding d. Setting Classes and Styles i. Using the Class Bindings ii. Using the Style Bindings e. Updating the Data in the Application 3. Using the Built-in Directives a. Preparing the Example Project b. Using the Built-in Directives i. Using the ngIf Directive ii. Using the ngSwitch Directive iii. Using the ngFor Directive iv. Using the ngTemplateOutlet Directive v. Using Directives Without an HTML Element c. Understanding One-Way Data Binding Restrictions i. Using Idempotent Expressions ii. Understanding the Expression Context 4. Using Events and Forms a. Preparing the Example Project i. Importing the Forms Module ii. Preparing the Component and Template b. Using the Event Binding i. Using Event Data ii. Handling Events in the Component iii. Using Template Reference Variables c. Using Two-Way Data Bindings i. Using the ngModel Directive d. Working with Forms i. Adding a Form to the Example Application ii. Adding Form Data Validation iii. Validating the Entire Form iv. Completing the Form **EVIDENCE GUIDANCE** Critical aspects of evidence: **Assessment must confirm the ability to –** • **create angular web application using Data Bindings, Built-in Directives, Events and Forms**	05/02/2025	02/03/2025

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
			Assessment of unit: This unit may be assessed by taking **minimum one (1) quiz** after completion of the unit. Each quiz will contain **10 questions of 1 mark each**. A **total mark of each quiz is 10**. **Quiz time is 10 minutes**.		
168	Module12	STC CS Angular-1205	**Creating Directives** 1. Creating Attribute Directives a. Preparing the Example Project b. Creating a Simple Attribute D i. Applying a Custom Di c. Accessing Application Data in a Directive i. Reading Host Element Attributes ii. Creating Data-Bound Input Properties iii. Responding to Input Property Changes d. Creating Custom Events i. Binding to a Custom Event e. Creating Host Element Bindings f. Creating a Two-Way Binding on the Host Element g. Exporting a Directive for Use in a Template Variable 2. Creating Structural Directives a. Preparing the Example Project b. Creating a Simple Structural Directive i. Implementing the Structural Directive Class ii. Enabling the Structural Directive iii. Using the Concise Structural Directive Syntax c. Creating Iterating Structural Directives i. Providing Additional Context Data ii. Using the Concise Structure Syntax iii. Dealing with Property-Level Data Changes iv. Dealing with Collection-Level Data Changes d. Querying the Host Element Content i. Querying Multiple Content Children ii. Receiving Query Change Notifications **EVIDENCE GUIDANCE** Critical aspects of evidence: **Assessment must confirm the ability to –** • **create angular attribute and structural directives** Assessment of unit: This unit may be assessed by taking **minimum one (1) quiz** after completion of the unit. Each quiz will contain **10 questions of 1 mark each**. A **total mark of each quiz is 10**. **Quiz time is 10 minutes**.	06/02/2025	03/03/2025
0			Exam Review Class	08/02/2025	
0			Exam	09/02/2025	

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>169</td><td>Module12</td><td>STC CS Angular-1206</td><td>
Components and Pipes
1. Understanding Components
a. Preparing the Example Project
b. Structuring an Application with Components
i. Creating New Components
ii. Defining Templates
iii. Completing the Component Restructure
c. Using Component Styles
i. Defining External Component Styles
ii. Using Advanced Style Features
d. Querying Template Content
2. Using and Creating Pipes
a. Preparing the Example Project
b. Understanding Pipes
c. Creating a Custom Pipe
i. Registering a Custom Pipe
ii. Applying a Custom Pipe
iii. Combining Pipes
iv. Creating Impure Pipes
d. Using the Built-in Pipes
i. Formatting Numbers
ii. Formatting Currency Values
iii. Formatting Percentages
iv. Formatting Dates
v. Changing String Case
vi. Serializing Data as JSON
vii. Slicing Data Arrays
viii. Formatting Key-Value Pairs
ix. Selecting Values
x. Pluralizing Values
xi. Using the Async Pipe
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create angular components and pipes</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>10/02/2025</td><td>04/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>170</td><td>Module12</td><td>STC CS Angular-1207</td><td>
Services and Service Providers
1. Using Services
a. Preparing the Example Project
b. Understanding the Object Distribution Problem
i. Demonstrating the Problem
ii. Distributing Objects as Services Using Dependency Injection
iii. Declaring Dependencies in Other Building Blocks
c. Understanding the Test Isolation Problem
i. Isolating Components Using Services and Dependency Injection
d. Completing the Adoption of Services
i. Updating the Root Component and Template
ii. Updating the Child Components
2. Using Service Providers
a. Preparing the Example Project
b. Using Service Providers
i. Using the Class Provider
ii. Using the Value Provider
iii. Using the Factory Provider
iv. Using the Existing Service Provider
c. Using Local Providers
i. Understanding the Limitations of Single Service Objects
ii. Creating Local Providers in a Component
iii. Understanding the Provider Alternatives
iv. Controlling Dependency Resolution
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create and use angular services and service providers</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>11/02/2025</td><td>05/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>171</td><td>Module12</td><td>STC CS Angular-1208</td><td>
Using and Creating Modules
1. Using and Creating Modules
a. Preparing the Example Project
b. Understanding the Root Module
i. Understanding the imports Property
ii. Understanding the declarations Property
iii. Understanding the providers Property
iv. Understanding the bootstrap Property
c. Creating Feature Modules
i. Creating a Model Module
ii. Creating a Utility Feature Module
iii. Creating a Feature Module with Components
2. Advanced Angular Features
a. Creating and Starting the Example Project
i. Adding and Configuring the Bootstrap CSS Package
ii. Creating the Project Structure
b. Creating the Model Module
i. Creating the Product Data Type
ii. Creating the Data Source and Repository
iii. Completing the Model Module
c. Creating the Messages Module
i. Creating the Message Model and Service
ii. Creating the Component and Template
iii. Completing the Message Module
d. Creating the Core Module
i. Creating the Shared State Service
ii. Creating the Table Component
iii. Creating the Form Component
iv. Completing the Core Module
e. Completing the Project
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• create and use angular modules</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>12/02/2025</td><td>06/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>172</td><td>Module12</td><td>STC CS Angular-1209</td><td>
Using the Forms API and Making HTTP Requests
1. Using the Forms API, Part 1
a. Preparing for This Chapter
b. Understanding the Reactive Forms API
c. Rebuilding the Form Using the API
i. Responding to Form Control Changes
ii. Managing Control State
iii. Managing Control Validation
iv. Adding Additional Controls
d. Working with Multiple Form Controls
i. Using a Form Group with a Form Element
ii. Accessing the Form Group from the Template
iii. Displaying Validation Messages with a Form Group
iv. Nesting Form Controls
2. Using the Forms API, Part 2
a. Preparing for This Chapter
b. Creating Form Components Dynamically
i. Using a Form Array
ii. Adding and Removing Form Controls
iii. Validating Dynamically Created Form Controls
iv. Filtering the FormArray Values
c. Creating Custom Form Validation
i. Creating a Directive for a Custom Validator
ii. Validating Across Multiple Fields
iii. Performing Validation Asynchronously
3. Making HTTP Requests
a. Preparing the Example Project
i. Configuring the Model Feature Module
ii. Creating the Data File
iii. Running the Example Project
b. Understanding RESTful Web Services
c. Replacing the Static Data Source
i. Creating the New Data Source Service
ii. Configuring the Data Source
iii. Using the REST Data Source
iv. Saving and Deleting Data
d. Consolidating HTTP Requests
e. Making Cross-Origin Requests
i. Using JSONP Requests
f. Configuring Request Headers
g. Handling Errors
i. Generating User-Ready Messages
ii. Handling the Errors
<table>
<tr><td colspan="2">EVIDENCE GUIDANCE</td></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• use Forms API
• make HTTP request</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>13/02/2025</td><td>10/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>173</td>
<td>Module12</td>
<td>STC CS Angular-1210</td>
<td>
Routing and Navigation

Routing and Navigation: Part 1
<ol type="a">
Preparing the Example Project
Getting Started with Routing
<ol type="i">
Creating a Routing Configuration
Creating the Routing Component
Updating the Root Module
Completing the Configuration
Adding Navigation Links
Understanding the Effect of Routing

Completing the Routing Implementation
<ol type="i">
Handling Route Changes in Components
Using Route Parameters
Navigating in Code
Receiving Navigation Events
Removing the Event Bindings and Supporting Code

Routing and Navigation: Part 2
<ol type="a">
Preparing the Example Project
<ol type="i">
Adding Components to the Project

Using Wildcards and Redirections
<ol type="i">
Using Wildcards in Routes
Using Redirections in Routes

Navigating Within a Component
<ol type="i">
Responding to Ongoing Routing Changes
Styling Links for Active Routes
Fixing the All Button

Creating Child Routes
<ol type="i">
Creating the Child Route Outlet
Accessing Parameters from Child Routes

Routing and Navigation: Part 3
<ol type="a">
Preparing the Example Project
Guarding Routes
<ol type="i">
Delaying Navigation with a Resolver
Preventing Navigation with Guards

Loading Feature Modules Dynamically
<ol type="i">
Creating a Simple Feature Module
Loading the Module Dynamically
Guarding Dynamic Modules

Targeting Named Outlets
<ol type="i">
Creating Additional Outlet Elements
Navigating When Using Multiple Outlets

<table>
<tr><td colspan="2">EVIDENCE GUIDANCE</td></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
create route and navigation using angular</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>15/02/2025</td>
<td>11/03/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>174</td><td>Module12</td><td>STC CS Angular-1211</td><td>
Animations and Component Libraries
1. Using Animations
a. Preparing the Example Project
i. Disabling the HTTP Delay
ii. Simplifying the Table Template and Routing Configuration
b. Getting Started with Angular Animation
i. Enabling the Animation Module
ii. Creating the Animation
iii. Applying the Animation
iv. Testing the Animation Effect
c. Understanding the Built-in Animation States
d. Understanding Element Transitions
i. Creating Transitions for the Built-in States
ii. Controlling Transition Animations
e. Understanding Animation Style Groups
i. Defining Common Styles in Reusable Groups
ii. Using Element Transformations
iii. Applying CSS Framework Styles
2. Working with Component Libraries
a. Preparing for This Chapter
i. Installing the Component Library
ii. Adjusting the HTML File
iii. Running the Project
b. Using the Library Components
i. Using the Angular Button Directive
ii. Using the Angular Material Table
c. Matching the Component Library Theme
i. Creating the Custom Component
ii. Using the Angular Material Theme
iii. Applying the Ripple Effect
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Using Animation
• Using Component Library</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>16/02/2025</td><td>12/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>175</td><td>Module13</td><td>STC CS React-1301</td><td>Getting Ahead in React and the Node Package Manager
1. Getting Ahead in React
a. React Primer
i. Component-Based Architecture
ii. Document Object Model (DOM)
iii. Client-Side Rendering and Server-Side Rendering
iv. Unidirectional Flow/One-Way Data Binding
b. Why React?
c. Career Opportunities in React
2. Getting Started with the Node Package Manager
a. Node Package Manager Overview
b. Node Version Manager
c. Installing Packages
i. Node Packaged Modules
ii. Package.json and Package-lock.json
iii. Semantic Versioning or Semver
d. Yarn
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• understanding Reat
• understanding Node Package Manager
• understanding Node Version Manager
• installing Node packages
• learning Yarn
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>17/02/2025</td><td>13/03/2025</td></tr>
<tr><td>176</td><td>Module13</td><td>STC CS React-1302</td><td>React Function Components & TypeScript and Setting Up an Enterprise-Level App
1. Getting Started with React Function Components and TypeScript
a. Creating a create-react-app App
b. Declaration Files for TypeScript with React
i. The Typings in TypeScript
c. Understanding React Function Components with Props
d. The Added IntelliSense in the Code Editor
e. React Hooks
i. React Hooks: useState
ii. React Hooks: useEffect
2. Setting Up an Enterprise-Level App
a. How to Select a Good Boilerplate
i. Pros of Using a Boilerplate
ii. Cons of Using a Boilerplate
b. Cloning a Boilerplate
c. React Boilerplate Meets CRA
i. Husky
ii. Dependencies
iii. A Closer Look at Some Dependencies
iv. Types Dependencies
d. Setting Up Your Visual Studio Code (Hybrid Text Editor)
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• learn React function components
• learning TypeScript
• learning React Hooks
• set-up an Enterprise-Level App
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>18/02/2025</td><td>15/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>177</td><td>Module13</td><td>STC CS React-1303</td><td>
Navigating React Router and Writing Local States, Sending HTTP Requests & ApexCharts

Navigating React Router

a. Why React Router?
b. Why Material-UI?
c. Getting Started
i. Basics Recap
d. Creating routes.tsx
e. Building a Dashboard
i. Navigation Bar
ii. Navigating React Router: <Redirect>
f. Creating the Dashboard Layout
i. Navigating Path: <Link>
ii. React Hook: useRouteMatch
g. Dashboard Component
i. Render Props
h. Going into React.lazy( )
i. React Suspense
ii. Eager Loading vs. Lazy Loading

Writing Local States, Sending HTTP Requests, and ApexCharts

a. Strict Type-Checking Options in TypeScript
b. Adding Fake Data Using json-server
c. Sending HTTP Requests Using Axios
i. Shaping the Object
ii. Making Requests with Axios
d. Installing ApexCharts
i. Creating the Main Layout
ii. Using React Feather Icons

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
navigate React Router
write Local States
send HTTP Requests
create chart using ApexCharts</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>19/02/2025</td><td>16/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>178</td><td>Module13</td><td>STC CS React-1304</td><td>Writing Data Tables, Formik Forms, and Yup Validations
1. Component Overview
a. Form Handling
b. Formik
c. Yup
2. Data Tables
a. Product Create View
b. Product Create Form
c. Product List View
3. Updating the Routes
4. Updating the Sidebar Dashboard
5. Creating a Collapsible Sidebar Menu
6. Cleaning Up a Bit
7. Defining Type Alias of ProductType
8. Creating the Products Endpoint
9. Creating the Products Service
a. Updating the db.json Data
b. Sending an HTTP Request
c. Refactoring the ProductListView
d. Creating Additional UI Styling
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> – write Data Tables; create and use Formik Forms; validating form using Yup Validations
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>20/02/2025</td><td>17/03/2025</td></tr>
<tr><td>179</td><td>Module13</td><td>STC CS React-1304</td><td>Writing Data Tables, Formik Forms, and Yup Validations: Part 2
10. Updating the ProductCreateView
11. Updating the ProductCreateForm
12. Installing React Quill
13. Installing React Dropzone
14. Installing a Notification Library
15. Updating Product Create Form
a. For Your Activity
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> – write Data Tables; create and use Formik Forms; validating form using Yup Validations
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>22/02/2025</td><td>18/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>180</td><td>Module13</td><td>STC CS React-1305</td><td>
Managing State Using Redux with Redux Toolkit
1. Managing State Using Redux with Redux Toolkit
a. Redux Overview
i. Why Use Redux?
b. Redux Toolkit
c. The Flow of Redux State Management
i. Using RTK in CodeSandbox
2. Setting Up Redux Toolkit and Dispatching an Asynchronous Action
a. Creating the Calendar View Component
i. Updating the Routes
ii. Updating the Dashboard Sidebar Nav
iii. Reducers
b. Cleanup Time
c. Updating Axios
d. Implementing Redux Toolkit
e. Non-asynchronous Actions/Synchronous Actions
f. Asynchronous Actions
g. Updating the Root Reducer
i. Updating the CalendarView
3. Creating, Deleting, and Updating Events on FullCalendar Using RTK
a. Installing Moment and FullCalendar
i. Updating the Root Component
ii. Updating the calendarSlice
iii. Updating the CalendarView
iv. Creating the Header Component
v. Updating the CalendarView
vi. Updating the Header
vii. Updating the CalendarView
viii. Creating the Toolbar
ix. Styling the CalendarView
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Manage state using redux with redux toolkit
• set up redux toolkit
• dispatch an asynchronous action
• create, delete, and update events using RTK
• Learn to use keyboard events</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>23/02/2025</td><td>19/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>181</td><td>Module13</td><td>STC CS React-1305</td><td>
Managing State Using Redux with Redux Toolkit
1. Managing State Using Redux with Redux Toolkit
a. Redux Overview
i. Why Use Redux?
b. Redux Toolkit
c. The Flow of Redux State Management
i. Using RTK in CodeSandbox
2. Setting Up Redux Toolkit and Dispatching an Asynchronous Action
a. Creating the Calendar View Component
i. Updating the Routes
ii. Updating the Dashboard Sidebar Nav
iii. Reducers
b. Cleanup Time
c. Updating Axios
d. Implementing Redux Toolkit
e. Non-asynchronous Actions/Synchronous Actions
f. Asynchronous Actions
g. Updating the Root Reducer
i. Updating the CalendarView
3. Creating, Deleting, and Updating Events on FullCalendar Using RTK
a. Installing Moment and FullCalendar
i. Updating the Root Component
ii. Updating the calendarSlice
iii. Updating the CalendarView
iv. Creating the Header Component
v. Updating the CalendarView
vi. Updating the Header
vii. Updating the CalendarView
viii. Creating the Toolbar
ix. Styling the CalendarView
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Manage state using redux with redux toolkit
• set up redux toolkit
• dispatch an asynchronous action
• create, delete, and update events using RTK
• Learn to use keyboard events</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>24/02/2025</td><td>20/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>182</td><td>Module13</td><td>STC CS React-1306</td><td>
Protecting Routes and Authentication in React

Setting Up the Fake Server
<ol type="a">
db.json
API: Login and Register
authService

Setting Up the Login Form
Creating a Register Form
Adding the Login Page
Updating the Routes
JSON Web Token (JWT)
Creating a ProtectedRoute Component
Updating the Routes.tsx
Updating the Dashboard Sidebar Navigation
Time to Test

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Protect routes
implement authentication in react</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>25/02/2025</td><td>22/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>183</td>
<td>Module13</td>
<td>STC CS React-1307</td>
<td>
Writing a Profile Form and Syncing It to Components
1. Writing a Profile Form and Syncing It to Components
a. Creating the Claims Type
i. Creating the User Type
b. Adding APIs: Users and UsersDb
i. Creating the userDbService
ii. Updating the authService
c. Another Way of Using Redux
i. Creating the profileActionTypes
ii. Creating the profileAsyncActions
iii. Creating the profileSlice
d. Adding profileReducer to the Reducers
e. Creating the authSlice
f. Adding the authSlice to the Reducers
i. Installing JWT-decode
ii. Updating the ProtectedRoute
g. Updating the Login Form
h. Updating the Register Form
i. Creating the Header Profile
i. Updating the Navigation Bar
j. Creating the AccountView
i. Adding Images
ii. Creating the Pricing Page
k. Updating the Routes
2. Updating the Dashboard Sidebar Navigation
a. Updating the db.json
b. Updating the Dashboard Sidebar Navigation
c. Creating the Yup Profile Validation
d. Creating the AccountView Page
e. Creating the Profile Details
f. Creating General Settings
g. Creating the Header
3. Creating the Notifications, Security, and Subscription Pages
a. Creating the Notifications Page
b. Creating the Security Page
c. Creating the Subscription Page
d. Updating the AccountView
e. Refresh
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• write a profile form and sync It to components</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>26/02/2025</td>
<td>23/03/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>184</td><td>Module13</td><td>STC CS React-1308</td><td>
Making the App Mobile-Friendly and Styling Methods for React Components and Deploying React in Netlify and in Docker

Making the App Mobile-Friendly
<ol type="a">
Getting Started
Updating a Package Library Safely
Updating the HomePage
Updating the About Page
Updating the Not Found Page
Making the Navbar Mobile-Friendly
Making the Dashboard Sidebar Navigation Mobile-Friendly
Making the Dashboard Layout Mobile-Friendly

Styling Methods for React Components
<ol type="a">
Inline
Plain CSS
Global
CSS Modules
CSS-in-JS
<ol type="i">
Styled-Components Library

Emotion in React

Deploying React in Netlify and in Docker
<ol type="a">
GitHub
Netlify
Docker
Docker Ignore
NGINX Config
<ol type="i">
Dockerfile
RUN npm install
Copy . ./
Docker Hub
Docker Commands

<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
Make mobile-friendly app
apply CSS and create CSS module
deploying react in Netlify and in Docker</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>27/02/2025</td><td>24/03/2025</td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>185</td>
<td>Module14</td>
<td>STC CS MAUI-1401</td>
<td>
Building Mobile and Desktop Apps Using .NET MAUI
1. Understanding XAML
a. Simplifying code using XAML
b. .NET MAUI namespaces
c. Type converters
d. Choosing between .NET MAUI controls
e. Markup extensions
2. Understanding .NET MAUI
a. Development tools for mobile first, cloud first
b. Installing .NET MAUI workloads manually
i. Using Windows to create iOS and macOS apps
c. .NET MAUI user interface components
i. Shell control
ii. ListView control
iii. Entry and Editor controls
d. .NET MAUI handlers
e. Writing platform-specific code
3. Building mobile and desktop apps using .NET MAUI
a. Creating a virtual Android device for local app testing
b. Enabling Windows developer mode
c. Creating a .NET MAUI solution
d. Adding shell navigation and more content pages
e. Implementing more content pages
4. Using shared resources
a. Defining resources to share across an app
b. Referencing shared resources
c. Changing shared resources dynamically
5. Using data binding
a. Binding to elements
6. Understanding MVVM
a. INotifyPropertyChanged interface
b. ObservableCollection class
c. Creating a view model with two-way data binding
d. Creating views for the customers list and customer details
e. Testing the .NET MAUI app
7. Consuming a web service from a mobile app
a. Creating a Minimal API web service for customers
b. Configuring the web service to allow unsecure requests
c. Connecting to local web services while testing
d. Configuring the iOS app to allow unsecure connections
e. Configuring the Android app to allow unsecure connections
f. Getting customers from the web service
8. Practicing and exploring
9. Summary
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Able to create cross platform desktop and mobile applications</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>01/03/2025</td>
<td>25/03/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>186</td>
<td>Module14</td>
<td>STC CS MAUI-1401</td>
<td>
Building Mobile and Desktop Apps Using .NET MAUI
1. Understanding XAML
a. Simplifying code using XAML
b. .NET MAUI namespaces
c. Type converters
d. Choosing between .NET MAUI controls
e. Markup extensions
2. Understanding .NET MAUI
a. Development tools for mobile first, cloud first
b. Installing .NET MAUI workloads manually
i. Using Windows to create iOS and macOS apps
c. .NET MAUI user interface components
i. Shell control
ii. ListView control
iii. Entry and Editor controls
d. .NET MAUI handlers
e. Writing platform-specific code
3. Building mobile and desktop apps using .NET MAUI
a. Creating a virtual Android device for local app testing
b. Enabling Windows developer mode
c. Creating a .NET MAUI solution
d. Adding shell navigation and more content pages
e. Implementing more content pages
4. Using shared resources
a. Defining resources to share across an app
b. Referencing shared resources
c. Changing shared resources dynamically
5. Using data binding
a. Binding to elements
6. Understanding MVVM
a. INotifyPropertyChanged interface
b. ObservableCollection class
c. Creating a view model with two-way data binding
d. Creating views for the customers list and customer details
e. Testing the .NET MAUI app
7. Consuming a web service from a mobile app
a. Creating a Minimal API web service for customers
b. Configuring the web service to allow unsecure requests
c. Connecting to local web services while testing
d. Configuring the iOS app to allow unsecure connections
e. Configuring the Android app to allow unsecure connections
f. Getting customers from the web service
8. Practicing and exploring
9. Summary
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Able to create cross platform desktop and mobile applications</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>02/03/2025</td>
<td>19/04/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>187</td>
<td>Module14</td>
<td>STC CS MAUI-1401</td>
<td>
Building Mobile and Desktop Apps Using .NET MAUI
1. Understanding XAML
a. Simplifying code using XAML
b. .NET MAUI namespaces
c. Type converters
d. Choosing between .NET MAUI controls
e. Markup extensions
2. Understanding .NET MAUI
a. Development tools for mobile first, cloud first
b. Installing .NET MAUI workloads manually
i. Using Windows to create iOS and macOS apps
c. .NET MAUI user interface components
i. Shell control
ii. ListView control
iii. Entry and Editor controls
d. .NET MAUI handlers
e. Writing platform-specific code
3. Building mobile and desktop apps using .NET MAUI
a. Creating a virtual Android device for local app testing
b. Enabling Windows developer mode
c. Creating a .NET MAUI solution
d. Adding shell navigation and more content pages
e. Implementing more content pages
4. Using shared resources
a. Defining resources to share across an app
b. Referencing shared resources
c. Changing shared resources dynamically
5. Using data binding
a. Binding to elements
6. Understanding MVVM
a. INotifyPropertyChanged interface
b. ObservableCollection class
c. Creating a view model with two-way data binding
d. Creating views for the customers list and customer details
e. Testing the .NET MAUI app
7. Consuming a web service from a mobile app
a. Creating a Minimal API web service for customers
b. Configuring the web service to allow unsecure requests
c. Connecting to local web services while testing
d. Configuring the iOS app to allow unsecure connections
e. Configuring the Android app to allow unsecure connections
f. Getting customers from the web service
8. Practicing and exploring
9. Summary
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Able to create cross platform desktop and mobile applications</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>03/03/2025</td>
<td>20/04/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr>
<td>188</td>
<td>Module14</td>
<td>STC CS MAUI-1401</td>
<td>
Building Mobile and Desktop Apps Using .NET MAUI
1. Understanding XAML
a. Simplifying code using XAML
b. .NET MAUI namespaces
c. Type converters
d. Choosing between .NET MAUI controls
e. Markup extensions
2. Understanding .NET MAUI
a. Development tools for mobile first, cloud first
b. Installing .NET MAUI workloads manually
i. Using Windows to create iOS and macOS apps
c. .NET MAUI user interface components
i. Shell control
ii. ListView control
iii. Entry and Editor controls
d. .NET MAUI handlers
e. Writing platform-specific code
3. Building mobile and desktop apps using .NET MAUI
a. Creating a virtual Android device for local app testing
b. Enabling Windows developer mode
c. Creating a .NET MAUI solution
d. Adding shell navigation and more content pages
e. Implementing more content pages
4. Using shared resources
a. Defining resources to share across an app
b. Referencing shared resources
c. Changing shared resources dynamically
5. Using data binding
a. Binding to elements
6. Understanding MVVM
a. INotifyPropertyChanged interface
b. ObservableCollection class
c. Creating a view model with two-way data binding
d. Creating views for the customers list and customer details
e. Testing the .NET MAUI app
7. Consuming a web service from a mobile app
a. Creating a Minimal API web service for customers
b. Configuring the web service to allow unsecure requests
c. Connecting to local web services while testing
d. Configuring the iOS app to allow unsecure connections
e. Configuring the Android app to allow unsecure connections
f. Getting customers from the web service
8. Practicing and exploring
9. Summary
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Able to create cross platform desktop and mobile applications</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td>
<td>04/03/2025</td>
<td>21/04/2025</td>
</tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>189</td><td>Module14</td><td>STC CS MAUI-1401</td><td>
Building Mobile and Desktop Apps Using .NET MAUI
1. Understanding XAML
a. Simplifying code using XAML
b. .NET MAUI namespaces
c. Type converters
d. Choosing between .NET MAUI controls
e. Markup extensions
2. Understanding .NET MAUI
a. Development tools for mobile first, cloud first
b. Installing .NET MAUI workloads manually
i. Using Windows to create iOS and macOS apps
c. .NET MAUI user interface components
i. Shell control
ii. ListView control
iii. Entry and Editor controls
d. .NET MAUI handlers
e. Writing platform-specific code
3. Building mobile and desktop apps using .NET MAUI
a. Creating a virtual Android device for local app testing
b. Enabling Windows developer mode
c. Creating a .NET MAUI solution
d. Adding shell navigation and more content pages
e. Implementing more content pages
4. Using shared resources
a. Defining resources to share across an app
b. Referencing shared resources
c. Changing shared resources dynamically
5. Using data binding
a. Binding to elements
6. Understanding MVVM
a. INotifyPropertyChanged interface
b. ObservableCollection class
c. Creating a view model with two-way data binding
d. Creating views for the customers list and customer details
e. Testing the .NET MAUI app
7. Consuming a web service from a mobile app
a. Creating a Minimal API web service for customers
b. Configuring the web service to allow unsecure requests
c. Connecting to local web services while testing
d. Configuring the iOS app to allow unsecure connections
e. Configuring the Android app to allow unsecure connections
f. Getting customers from the web service
8. Practicing and exploring
9. Summary
<table>
<tr><th colspan="2">EVIDENCE GUIDANCE</th></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Able to create cross platform desktop and mobile applications</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>05/03/2025</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>190</td><td>Module14</td><td>STC CS MAUI-1402</td><td>Integrating .NET MAUI Apps with Blazor and Native Platforms
1. Building .NET MAUI Blazor Hybrid apps
a. Creating a .NET MAUI Blazor project
b. Adding a shell and .NET MAUI pages
c. Creating a minimal API web service for categories
d. Configuring the .NET MAUI app to allow unsecure connections
e. Implementing the Model-View-ViewModel pattern
f. Getting categories from the web service
2. Integrating with native platforms
a. Working with the system clipboard
b. Picking files from the local filesystem
i. Enabling media and file picking on Windows
ii. Enabling media and file picking on Android
iii. Enabling media and file picking on iOS
iv. Integrating with the media and file picker
c. Creating new windows
d. Getting device information
i. Enabling device information on Android
ii. Adding device information to the app
iii. Integrating with desktop menu bars
iv. Popping up a toast notification
3. Using third-party control libraries
4. Practicing and exploring
5. Summary
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> –
• Able to integrating .NET MAUI apps with Blazor and native platforms
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>06/03/2025</td><td></td></tr>
<tr><td>0</td><td></td><td></td><td>Exam Review Class</td><td>08/03/2025</td><td></td></tr>
<tr><td>0</td><td></td><td></td><td>Exam</td><td>09/03/2025</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>191</td><td>Module14</td><td>STC CS MAUI-1402</td><td>
Integrating .NET MAUI Apps with Blazor and Native Platforms
1. Building .NET MAUI Blazor Hybrid apps
a. Creating a .NET MAUI Blazor project
b. Adding a shell and .NET MAUI pages
c. Creating a minimal API web service for categories
d. Configuring the .NET MAUI app to allow unsecure connections
e. Implementing the Model-View-ViewModel pattern
f. Getting categories from the web service
2. Integrating with native platforms
a. Working with the system clipboard
b. Picking files from the local filesystem
i. Enabling media and file picking on Windows
ii. Enabling media and file picking on Android
iii. Enabling media and file picking on iOS
iv. Integrating with the media and file picker
c. Creating new windows
d. Getting device information
i. Enabling device information on Android
ii. Adding device information to the app
iii. Integrating with desktop menu bars
iv. Popping up a toast notification
3. Using third-party control libraries
4. Practicing and exploring
5. Summary
<table>
<tr><td colspan="2">EVIDENCE GUIDANCE</td></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Able to integrating .NET MAUI apps with Blazor and native platforms</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>10/03/2025</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>192</td><td>Module14</td><td>STC CS MAUI-1402</td><td>
Integrating .NET MAUI Apps with Blazor and Native Platforms
1. Building .NET MAUI Blazor Hybrid apps
a. Creating a .NET MAUI Blazor project
b. Adding a shell and .NET MAUI pages
c. Creating a minimal API web service for categories
d. Configuring the .NET MAUI app to allow unsecure connections
e. Implementing the Model-View-ViewModel pattern
f. Getting categories from the web service
2. Integrating with native platforms
a. Working with the system clipboard
b. Picking files from the local filesystem
i. Enabling media and file picking on Windows
ii. Enabling media and file picking on Android
iii. Enabling media and file picking on iOS
iv. Integrating with the media and file picker
c. Creating new windows
d. Getting device information
i. Enabling device information on Android
ii. Adding device information to the app
iii. Integrating with desktop menu bars
iv. Popping up a toast notification
3. Using third-party control libraries
4. Practicing and exploring
5. Summary
<table>
<tr><td colspan="2">EVIDENCE GUIDANCE</td></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Able to integrating .NET MAUI apps with Blazor and native platforms</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>11/03/2025</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>193</td><td>Module14</td><td>STC CS MAUI-1402</td><td>
Integrating .NET MAUI Apps with Blazor and Native Platforms
1. Building .NET MAUI Blazor Hybrid apps
a. Creating a .NET MAUI Blazor project
b. Adding a shell and .NET MAUI pages
c. Creating a minimal API web service for categories
d. Configuring the .NET MAUI app to allow unsecure connections
e. Implementing the Model-View-ViewModel pattern
f. Getting categories from the web service
2. Integrating with native platforms
a. Working with the system clipboard
b. Picking files from the local filesystem
i. Enabling media and file picking on Windows
ii. Enabling media and file picking on Android
iii. Enabling media and file picking on iOS
iv. Integrating with the media and file picker
c. Creating new windows
d. Getting device information
i. Enabling device information on Android
ii. Adding device information to the app
iii. Integrating with desktop menu bars
iv. Popping up a toast notification
3. Using third-party control libraries
4. Practicing and exploring
5. Summary
<table>
<tr><td colspan="2">EVIDENCE GUIDANCE</td></tr>
<tr><td>Critical aspects of evidence</td><td><u>Assessment must confirm the ability to</u> –
• Able to integrating .NET MAUI apps with Blazor and native platforms</td></tr>
<tr><td>Assessment of unit</td><td>This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td></tr>
</table>
</td><td>12/03/2025</td><td></td></tr>
</table>

<table>
<tr><th>Sl</th><th>Module</th><th>Competency Unit</th><th>Elements</th><th>Schedule Date</th><th>Actual Date</th></tr>
<tr><td>194</td><td>Module14</td><td>STC CS MAUI-1402</td><td>Integrating .NET MAUI Apps with Blazor and Native Platforms
1. Building .NET MAUI Blazor Hybrid apps
a. Creating a .NET MAUI Blazor project
b. Adding a shell and .NET MAUI pages
c. Creating a minimal API web service for categories
d. Configuring the .NET MAUI app to allow unsecure connections
e. Implementing the Model-View-ViewModel pattern
f. Getting categories from the web service
2. Integrating with native platforms
a. Working with the system clipboard
b. Picking files from the local filesystem
i. Enabling media and file picking on Windows
ii. Enabling media and file picking on Android
iii. Enabling media and file picking on iOS
iv. Integrating with the media and file picker
c. Creating new windows
d. Getting device information
i. Enabling device information on Android
ii. Adding device information to the app
iii. Integrating with desktop menu bars
iv. Popping up a toast notification
3. Using third-party control libraries
4. Practicing and exploring
5. Summary
EVIDENCE GUIDANCE
Critical aspects of evidence: <u>Assessment must confirm the ability to</u> – • Able to integrating .NET MAUI apps with Blazor and native platforms
Assessment of unit: This unit may be assessed by taking <u>minimum one (1) quiz</u> after completion of the unit. Each quiz will contain <u>10 questions of 1 mark each</u>. A <u>total mark of each quiz is 10</u>. <u>Quiz time is 10 minutes</u>.</td><td>13/03/2025</td><td></td></tr>
<tr><td>195</td><td>Module15</td><td>STC CS FCP-1501</td><td>Integration, Testing and Deployment
1. Integration, Testing and Deployment
a. Integrate Backend Web API with Frontend Client Web Application
b. Test and Fix Errors & Exceptions
c. Test & Fix Bugs
d. Test UI/UX
e. Test Performance & Optimize
f. Test & Select Best Deployment Technique
g. Deploy the Project on Windows Azure Cloud</td><td>15/03/2025</td><td></td></tr>
<tr><td>196</td><td>Module15</td><td>STC CS FCP-1501</td><td>Integration, Testing and Deployment
1. Integration, Testing and Deployment
a. Integrate Backend Web API with Frontend Client Web Application
b. Test and Fix Errors & Exceptions
c. Test & Fix Bugs
d. Test UI/UX
e. Test Performance & Optimize
f. Test & Select Best Deployment Technique
g. Deploy the Project on Windows Azure Cloud</td><td>16/03/2025</td><td></td></tr>
</table>

Sl	Module	Competency Unit	Elements	Schedule Date	Actual Date
197	Module15	STC CS FCP-1501	**Integration, Testing and Deployment** 1. Integration, Testing and Deployment a. Integrate Backend Web API with Frontend Client Web Application b. Test and Fix Errors & Exceptions c. Test & Fix Bugs d. Test UI/UX e. Test Performance & Optimize f. Test & Select Best Deployment Technique g. Deploy the Project on Windows Azure Cloud	17/03/2025	
0			**Interview Preparation**: • Learn interview questions from given books and materials for interview preparation. • Practice code from given code books and code materials.	18/03/2025	
0			**Interview Preparation**: • Learn interview questions from given books and materials for interview preparation. • Practice code from given code books and code materials.	19/03/2025	
0			**Interview Preparation**: • Learn interview questions from given books and materials for interview preparation. • Practice code from given code books and code materials.	20/03/2025	
0			Ramadan Make Up Review Class	05/04/2025	
0			Ramadan Make Up Review Class	06/04/2025	
0			Ramadan Make Up Review Class	07/04/2025	
0			Ramadan Make Up Review Class	08/04/2025	
0			Consultant instructation Class	09/04/2025	
0			Exam Review Class	15/04/2025	
0			Exam	16/04/2025	
0			Consultant instructation Class	17/04/2025	

+88 02 9132568, +88 02 9132576

IsDB-BISEW IT Scholarship Programme (https://idb-bisew.org/)

© 2017 Project Information System. All Rights Reserved | Powered By Show & Tell Consulting Ltd

(https://www.facebook.com/IDBBISEWSCHOLARSHIP/)

(https://plus.google.com/u/0/communities/107132260182543513057)

(https://www.youtube.com/channel/UCpkDA8qqR7tfvvTeXna5mkw)